University of Cape Town
Faculty of Science

Department of Mathematics and Applied Mathematics

Applied Mathematics Masters Thesis

submitted for the degree of

Master of Science

Biologically Motivated Reinforcement Learning in Spiking Neural Networks

by

Dean Rance
Student Number: RNCDEA001

Submission Date: March 1, 2022
Supervisor: Dr Jonathan Shock

Declaration of Authorship

I, DEAN RANCE, declare that this thesis titled, 'BIOLOGICALLY MOTIVATED REINFORCEMENT LEARNING IN SPIKING NEURAL NETWORKS' and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed:

Date:

Abstract

I consider the problem of Reinforcement Learning (RL) in a biologically feasible neural network model, as a proxy for investigating RL in the brain itself. Recent research has demonstrated that synaptic plasticity in the higher regions of the brain (such as the cortex and striatum) depends on neuromodulatory signals which encode, amongst other things, a response to reward from the environment. I consider which forms of synaptic plasticity rules might arise under the guidance of an Evolutionary Algorithm (EA), when an agent is tasked with making decisions in response to noisy stimuli (perceptual decision making). By proposing a general framework which captures many proposed biologically feasible phenomenological synaptic plasticity rules, including classical Spike-Time-Dependent Plasticity (STDP) rules and the triplet rules, and rate-based rules such as Oja's Rule and BCM rules, as well as their reward-modulated extensions (such as Reward-Modulated Spike-Time-Dependent Plasticity (R-STDP)), I allow a general biologically feasible neural network the ability to evolve the rules best suited for learning to solve perceptual decision-making tasks.

Acknowledgments

> Physics isn't the most important thing.
> Love is.
>
> Richard P. Feynman

In these times, I could fill this thesis with the emotional and moral support I have received during the studying for and creating of this project. My academic supervisor, Jonathan Shock, deserves the utmost gratitude not only for academic support but also for guidance in life and importantly the encouragement to take breaks when I needed them. He has also encouraged and endured my rather excessive approach to investigating and exploring the irrelevant tangentials.

For having people to learn the tools of this trade, both with and from, as well as for providing advice and helping me to figure out the details at various points, I would particularly like to thank Michael Perreira, Cristopher Currin, Sadiq Adewale Adedayo, William Podlaski and Eric DeWitt. For the efforts of making neuroscience open and accessible, the World Wide Neuro team and Neuromatch team have provided unparalleled resources for learning which, while not personal, were nonetheless significant. For sending me on this path in the first place, I would also like to express my gratitude towards the organisers of the IBRO-Simons Computational Neuroscience Imbizo and friends and colleagues from that Imbizo, and towards Xiao-Jing Wang and Rafael Bogacz for being accessible and helping me to build on their work.

The support of my friends and family has been indispensable. No person is an island, especially in these times. I cannot name you all, but I could not have done this without you.

Contents

Abbreviations

A-A All-to-All. 19, 29, 91

AdEx Adaptive Exponential LIF. 23, 92, 94

AIC Akaike Information Criterion. 87

AMPA α-amino-3-hydroxy-5-methyl-4-isoxazolepropionic acid. 21, 23, 24, 70, 73, 105, 112

AMPAR AMPA Receptor. 23, 49, 103

AP Action Potential. 10, 22, 23, 25, 29, 30, 37, 93

BPAP Backpropagating Action Potential. 22, 33

CMA-ES Covariance Matrix Adaptation Evolution Strategy. 5, 12–14, 19, 20, 50, 54, 56, 57, 62–65, 69, 73, 79–81, 89, 91, 95, 100, 111

DD Drift Diffusion. 18, 43–46, 48, 92

DE Differential Evolution. 53, 88

EA Evolutionary Algorithm. 3, 14, 18–20, 44, 50–55, 57, 61–63, 67, 69, 73, 77, 79, 86, 88, 91, 92, 97

EP Evolutionary Programming. 50, 54

EPANN Evolved Plastic Artificial Neural Network. 57

EPSC Excitatory Postsynaptic Current. 11, 34

ES Evolution Strategy. 50, 52–54, 56, 57

GA Genetic Algorithm. 50–52, 54

GABA γ-aminobutyric acid. 22, 24, 70, 105, 112

IF Integrate-and-Fire. 23

ISI Interspike Interval. 11, 37, 38, 103

iSTDP Inhibitory STDP. 10, 29, 93

LIF Leaky Integrate-and-Fire. 11, 15, 21, 23, 25, 26, 35, 37, 38, 42, 49, 94, 102, 112

LIP Lateral Intraparietal. 19, 43–45, 47

LNP Linear-Nonlinear Poisson. 95

LTD Long-Term Depression. 10, 11, 20, 26, 28, 29, 34, 35, 42, 93, 96, 97, 99

LTP Long-Term Potentiation. 10, 11, 20, 26, 28, 29, 33–35, 42, 93, 96, 97, 99

MSPRT Multisequential Probability Ratio Test. 46

MT Middle Temporal. 44, 45, 47

N-N Nearest Neighbours. 10, 29, 30, 32, 36

NEAT NeuroEvolution of Augmenting Topoligies. 57, 95

NES Natural Evolution Strategies. 57

NMDA N-methyl-D-aspartate. 15, 21, 23, 25, 47, 70, 73, 86, 88, 98, 103, 105, 106, 110, 112

NMDAR NMDA Receptor. 12, 14, 23, 24, 47–49, 70, 72, 77, 88, 90, 98, 105, 106

PSC Postsynaptic Current. 25, 93

PSP Postsynaptic Potential. 25, 93, 98

R-STDP Reward-Modulated Spike-Time-Dependent Plasticity. 3, 41, 46, 49

RCP Rapid Compensatory Process. 38

RDM Random Dot Motion. 5–7, 11–13, 19, 43, 44, 46, 47, 61–67, 79, 83, 88

RL Reinforcement Learning. 3, 17, 19, 40, 41, 50, 56, 57, 85, 97, 100

RPE Reward Prediction Error. 17, 97, 99

RT reaction time. 11, 44–46, 92

SNN Spiking Neural Network. 15, 18–21, 34, 85–88, 93, 102

SPRT Sequential Probability Ratio Test. 45

SRM Spike Response Model. 42, 95

STDP Spike-Time-Dependent Plasticity. 3, 5, 10, 11, 19, 26, 28–35, 41, 67, 68, 85, 86, 91–93, 100

STP Short-Term Plasticity. 25, 93

VTA Ventral Tegmental Area. 17, 99

List of Figures

Chapter 1

Introduction

> My son, be warned! Neither soar to high,
> lest the sun melt the wax; nor swoop too
> low, lest the feathers be wetted by the sea
>
> Daedalus to Icarus, as told by Robert Graves in The Greek Myths volume 1, [1]

Research within the past few years has begun to uncover the myriad of ways by which neuromodulatory signals or changes in the concentration of neuromodulators at the synapse can alter or influence the plasticity dynamics dependent on timing, synapse type, plasticity induction protocol and neuromodulator type [2, 3, 4, 5]. As the number of potential interactions grows, so too does the amount of experimental exploration required to investigate these phenomena. Such experimentation might be better guided with a theory-first approach, whereby normative descriptions of how such interactions should appear can be posited and confirmed or rejected.

One such signal thought to be correlated with neuromodulatory activity is reward in the form of Reward Prediction Error (RPE) signals encoded in the dopaminergic activity of Ventral Tegmental Area (VTA) neurons [6, 7, 8, 9]. What this means is that theoreticians can postulate about dopamine-modulated synaptic plasticity from a normative standpoint by combining Reinforcement Learning (RL) with biophysically inspired models of neural activity. One can ask, "Given what we know about RL and maximising reward, how *should* dopamine modulate plasticity?"

A normative approach to biology often receives criticism [10]. However, aside from suggesting further experiments, there are at least two (albeit related) ways in which an optimisation approach to fitting a biological model can offer insight, assuming the function being optimised for, the *fitness function*, is biologically relevant: it can provide a prior distribution over the true state of a biological phenomenon [10], and it can provide a regularising prior for fitting a model meant to

describe such a phenomenon [11].

A step in this process is the optimisation procedure itself. On deterministic optimisation problems Evolutionary Algorithms (EAs) are global optimisers in that given enough time, when mindful of introducing sufficient randomness and meeting fairly liberal criteria such as elitism, they are guaranteed to find a global optimum [12, 13]. However, just as importantly, they are incredibly versatile and can easily be implemented for optimisation of fitness functions which are non-differentiable or difficult to define or compute [13]. Their primary drawback may be the many (highly parallelisable) evaluations of the fitness function which need to be performed [14].

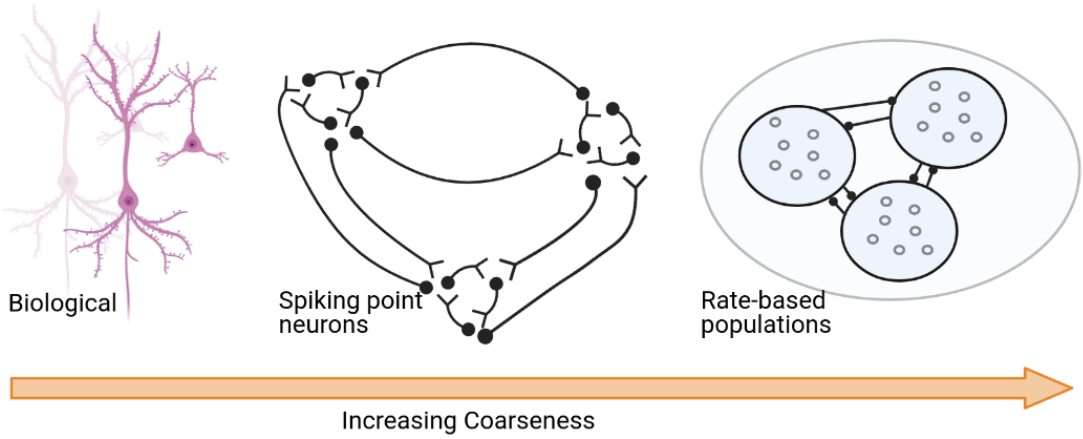


Figure 1.1: Neural activity can be modelled at various scales. One can model the geometry of the neuron, treat each neuron as being composed of various compartments or being point-neurons, or even model the rates of populations of neurons directly. Image created with BioRender.com software.

Yet a biologically inspired[1] neural network model can come in various forms, from the truly biophysical models incorporating cell structure and ion concentrations, through the coarser biologically inspired multicompartment or point-neuron Spiking Neural Networks (SNNs) with realistic membrane potential dynamics, to population-level rate-based models. Generally as one increases the coarseness of the model, one can also increase the efficiency at which the dynamics are simulated. Biophysical models of synaptic plasticity, on the other hand, require information about the synaptic dynamics usually excluded from these larger network models (such as calcium ion concentration at the synapses [15]). As such, when considering plasticity in a coarser biologically inspired model, one turns to phenomenological models of plasticity [16]. These models may posit abstract variables, such as "synaptic tags", with little to no attempt to identify them with any underlying biomolecular substrate or process.

One such family of biologically inspired neural network models are attractor decision making models, or recurrent neural circuit models [17]. These models can be seen to generalise the Drift

[1]One should be careful to use "biologically inspired" or "biophysically inspired" phrasal nomenclature. The adjective "biological" is reserved for models which capture a higher degree of biological realism such as spatial extent, ion currents, temperature of the extracellular fluid, etc. In a sense, the neurons of a biologically inspired SNN are to the study of actual cellular neurons as spherical cows in a vacuum are to dairy farming.

Diffusion (DD) models of decision making [18] for perceptual decision making tasks such as the Random Dot Motion (RDM) task, while also providing a potentially better fit to data and a biological interpretation of their components [17]. They effectively bridge the gap between psychological phenomenology and neural dynamics.

In this thesis I will use one such model, which I will call the Wang model, originally having been developed to account for persistently increased neural activity under neuromodulation [19] but adapted to explain firing rates of neural populations in the Lateral Intraparietal (LIP) area in the brains of monkeys performing the RDM task [20]. This model will be used to implement the action selection step, allowing the network to engage with the environment in pursuit of rewards. The family of tasks I will consider is an artificial version of the RDM task parameterised by coherence, and the fitness function will be determined by the number of correct successive trials on the RDM task. Performance on this task will be optimised by using Covariance Matrix Adaptation Evolution Strategy (CMA-ES) - an EA which performs well in continuous domain optimisation [21] - to determine independently both the synaptic weights of the model, and the plasticity rule driving synaptic weight changes. Avoiding potentially nonsensical questions about the interactions between biological substrates and abstract concepts, I will only consider the affects of an abstract reward variable R on the synaptic weight dynamics. I will use a coarse rate-based population level model, but one which faithfully represents a biophysically inspired SNN, to account for the many simulations the EA will run. In turn, building on the work done by [22, 23], I will use the Volterra extension to derive a family of Spike-Time-Dependent Plasticity (STDP) plasticity rules capturing many of the features typically included in such phenomenological models of plasticity, and under the assumption of inhomogeneous independent Poisson-like dynamics, determine the rate-based analogue of this family of rules which can be used to extend the rate-based Wang model to include plasticity dynamics.

The research topic of this thesis is to ascertain whether and how an EA can be used to determine biophysically inspired phenomenological synaptic plasticity dynamics capable of solving RL tasks when implemented in a biophysically inspired recurrent neural network model of decision making.

Some other work has used similar methods in various forms to those presented here. In [24] the Wang model is combined with a reward-driven plasticity rule which produces matching-law behaviour. However the plasticity rule considered was stochastic with discrete synapses, and thus not within the collection of STDP rules I considered. In [25] the Volterra framework for All-to-All (A-A) STDP rules was combined with CMA-ES, very similar to this work, but on different network topologies in an attempt to recover common unsupervised plasticity rules. Their work further considered inhibitory plasticity, but did not consider reward-driven plasticity.

Indeed most work I reviewed considered in isolation either unsupervised or reward-driven plasticity, with the notable exception of [26]. Their method of combining these two forms of plasticity was functionally distinct from my own, of incorporating an adjustable parameter β. In their model,

they implement independent eligibility traces for Long-Term Depression (LTD) and Long-Term Potentiation (LTP) which can be differentially influenced by dopamine concentrations. In the absence of a reward signal, these traces can still induce plasticity. In [27, 28] it is shown that there may be distinct eligibility traces for LTD and LTP (discussed below).

There has also been work on using EAs to fit SNNs to supervised learning problems. A comprehensive review of the innumerable times this has been tried cannot be provided, but a few insightful cases can be mentioned. In both [29] and [30] an SNN is fitted to solve a classification task. In the former they outline that any second-generation/analogue neural network can approximate any continuous function on a bounded domain, known as the Universal Approximation Theorem, and moreover that any such neural network of n neurons can be approximated arbitrarily well with $n + c$ spiking neurons in an SNN where c is a small number, implying a universal approximation property for SNNs.

There are also non-biological examples which nonetheless may inspire further work. In [31] a neuromodulation-dependent plasticity rule was evolved alongside the network topology, but little attempt is made to keep the model biophysically inspired, while [32] considers a neuromodulation-driven metalearning approach where a neuromodulatory network gates the plasticity in another prediction network. Other work on metalearning has considered learning the update rule for unsupervised learning, which is encoded by an artificial neural network, via stochastic gradient descent on the performance of the learnt unsupervised representations of data on later semi-supervised tasks [33, 34, 35]. This work provides a potentially alternative approach to determining the learning rule (via backpropagation and stochastic gradient descent) and it may be interesting to compare the resultant neural network with anatomy; however, they omit reinforcement or neuromodulatory signals, and thus their learning rules are not akin to the three-factor learning rules discussed herein.

In what follows, Chapter 2 provides a background of the relevant literature on the topics discussed, including phenomenological models of synaptic plasticity, models of decision making, and EAs as well as justification for an optimisation-driven approach. Chapter 3 provides the extension to the plasticity rules, and discusses implementation details of the task and network simulations. The results of the experiments with CMA-ES are discussed in Chapter 4, where it is observed that the evolved plasticity rules for different coherence values are significantly different. Chapter 5 discusses potential extensions to this work, alongside limitations. Finally, Chapter 6 concludes.

The Wang model itself, both the original SNN as well as its reduction via mean-field theory to a rate-based model, can be found in several standard texts including [19] and [36]. However, I have included both in the Appendix A.

Chapter 2

Literature Review

> When an axon of cell A is near enough to excite a cell B and repeatedly or persistently takes part in firing it, some growth process or metabolic change takes place in one or both cells such that A's efficiency, as one of the cells firing B, is increased.
>
> Donald Hebb, [37]

Here I will discuss the background and literature on the various topics brought together in this project, most notably synaptic plasticity, but also decision making, evolutionary algorithms, and ideas about normative approaches to biology (in this order). For reference, neurons and synapses are discussed first.

2.1 Background

The aim of this section is simply to describe the neurons of a Leaky Integrate-and-Fire (LIF) SNN model, as well as the currents arising from the conductance-based synapses, sufficiently to understand the Wang model as well as ideas of synaptic plasticity.

The main idiosyncrasies of the synapses discussed here are this: charge is not modeled as immediately added to the postsynaptic membrane potential as is sometimes done (see for examples [38]), but rather occurs over time due to the fraction of open synaptic channels following their own dynamics. Moreover, two sources of excitatory input are considered: α-amino-3-hydroxy-5-methyl-4-isoxazolepropionic acid (AMPA) mediated inputs and N-methyl-D-aspartate (NMDA) mediated

inputs, where the latter has a nonlinear voltage-dependent conductance given by the Jahr-Stevens formula (2.38) [39]. Inhibitory inputs are mediated by γ-aminobutyric acid (GABA).

2.1.1 Neurons

Neurons maintain a membrane potential V_i (where i is the neuron index), which, when rising above a threshold V_{thr}, leads to a cascade of biomolecular activity resulting in an Action Potential (AP), or spike [40]. We say the neuron *fires* an AP. The characteristic feature of an AP is the propagation of an electrical signal along the neuron's axon inducing the release of a neurotransmitter into the synaptic clefts between the axon and the cell bodies (usually on the dendrites) of the postsynaptic neurons. The neurotransmitters are biomolecules which bind to receptor proteins in the membrane of the postsynaptic cell. This binding leads - either directly or indirectly - to the opening of ion channels in the postsynaptic cell membrane, through which charged ions flow following their electrochemical gradient and thus changing the membrane potential of the postsynaptic cell. Importantly, the ion channels are selective to specific ions, meaning that the binding of specific transmitters to specific receptors can selectively increase or decrease the charge of the postsynaptic cell depending on the charge of the ion and the relative concentration of the ion within and without the cell.

When the AP is fired, it also propagates back into the dendritic arbor of the cell, called a Backpropagating Action Potential (BPAP). This BPAP can signal to synapses on the dendrites that the cell has recently fired.

At its baseline, the cell membrane is already polarised. Any phenomenon which causes the cell to depolarise towards its threshold V_{thr} is called *depolarising* or *excitatory*. Conversely, anything effecting an increase in polarisation is called *hyperpolarising* or *inhibitory*.

In the Wang model used in this thesis, there are two principle types of cells: pyramidal cells, exerting an excitatory influence on their postsynaptic targets, and inhibitory interneurons. Many of the parameters discussed below depend on the cell type. There are many more types of neurons in the brain but the grouping of neurons as being either excitatory or inhibitory is fairly general with few neurons belonging in both groups, an observation known as Dale's Law [41].

We can model the trajectory of the membrane potential of the neuron i with differential equation [38]

$$C_m \frac{dV_i(t)}{dt} = -g_m(V_i(t) - V_L) - I_i(t) \tag{2.1}$$

where C_m is the membrane capacitance, g_m is the membrane leak conductance (describing the permeability of the membrane) and $I_i(t)$ is the positive current flowing out of the cell (equivalently, negative current flowing in).

This can also be written with the membrane time constant $\tau_m = C_m/g_m$:

$$\tau_m \frac{dV_i(t)}{dt} = -(V_i(t) - V_L) - \frac{I_i(t)}{g_m} \tag{2.2}$$

Having a single variable for membrane potential per neuron implies that the neurons are of the point-neuron type. One could add components, in which case $V_i(t)$ inherits a spatial partial derivative describing current flow between components [38], but that is not done here.

In Integrate-and-Fire (IF) neurons we model the firing of an AP and resetting and refraction of the cell by adding the resetting rule $V_i(t) \leftarrow V_{reset}$ whenever $V(t)$ crosses the threshold V_{thr} from below [38]. After this the membrane potential remains at V_{thr} for a time τ_{refrac} describing the absolute refractory time of the neuron. The crossing or resetting times of the membrane potential are captured by the variable $S_i(t)$, called the spike train. That is,

$$S_i(t) := \sum_f \delta(t - t_i^f) \tag{2.3}$$

where the sum is taken over the firing times t_i^f of the cell, and δ is the Dirac-δ function. Combined with (2.1) or (2.2), this gives the LIF model. Other IF models do exist, such as the Adaptive Exponential LIF (AdEx) model [42, 43] which includes an exponential in its membrane dynamics and allows for an adaptive threshold and hence better captures the subthreshold membrane potential, or the Izhikevich neurons [44] which use polynomials for computational efficiency but can also capture qualitatively different spiking behaviour. However, in this project I will only use the simpler LIF model.

I will use subscripts such as i or j to index neurons, with the convention that j is the presynaptic neuron and i is the postsynaptic neuron. When considering a focal synapse between two neurons, such as when looking at the Volterra series for the plasticity dynamics as in (2.10), to be succinct I will use the notation of $X(t) = S_j(t)$ for the presynaptic spike train and $Y(t) = S_i(t)$ for the postsynaptic spike train, following the convention of [22].

2.1.2 Synapses

The Wang model considers the two most common neurotransmitters and three receptor types [19]. The neurotransmitter glutamate has an excitatory effect inducing AMPA to bind to the fast-acting AMPA Receptor (AMPAR) and by inducing NMDA to bind to the slower acting NMDA Receptor (NMDAR) [40]. Magnesium Mg^{2+} can bind to sites on the NMDARs restricting ion flow through the channel effectively reducing the average conductance of all the NMDARs at the synapse, but these ions can be dislodged by increasing the membrane potential. In this way the NMDAR ion channels are voltage dependent.

The neurotransmitter GABA binds to the relatively fast-acting $\mathrm{GABA_A}$ receptors and has an inhibitory effect on the postsynaptic cell.

At each synapse ij we have the fraction $s_{j,rec}$ of open channels of each type of receptor rec with dynamics dependent only on the presynaptic neuron j. Each receptor type also has a corresponding conductance $g_{rec}(V_i)$ which may (for NMDAR) or may not (for AMPA and $\mathrm{GABA_A}$ receptors) depend on membrane potential. Finally, each synapse has its synaptic strength w_{ij} which scales the relative conductances of the cells. It is this variable w_{ij} which undergoes change, or synaptic plasticity.

To determine the current I_i out of the postsynaptic cell, we also need to know the *driving force*. This is given by the difference between the membrane potential and the reversal potentials of the excitatory (V_E) and inhibitory (V_I) effects. Thus we arrive at, for the contribution of a single presynaptic neuron j,

$$\begin{aligned} I_i(t) = w_{ij}\, g_{NMDA}(V_i(t))\, s_{j,NMDA}(t)\, (V_i(t) - V_E) \\ + w_{ij}\, g_{AMPA}\, s_{j,AMPA}(t)\, (V_i(t) - V_E) \end{aligned} \tag{2.4}$$

if the presynaptic cell is a pyramidal (glutamatergic) cell, or

$$I_i(t) = w_{ij}\, g_{GABA}\, s_{j,GABA}(t)\, (V_i(t) - V_I) \tag{2.5}$$

if the presynaptic cell is an inhibitory interneuron (GABAergic). This dependence on the driving forces, $(V_i(t) - V_{E/I})$, means that the model is of the conductance-based type. The total input current $I_i(t)$ becomes a summation over all presynaptic neurons with indices j in a collection $\mathcal{P}$:

$$\begin{aligned} I_{AMPA}(t) + I_{NMDA}(t) &:= (V(t) - V_E) \sum_{j \in \mathcal{P}} w_j \left[g_{AMPA} \cdot s_{j,AMPA}(t) + g_{NMDA}(V(t)) \cdot s_{j,NMDA}(t) \right], \\ I_{GABA}(t) &:= (V(t) - V_I) \sum_{j \in \mathcal{P}} w_j \cdot g_{GABA} \cdot s_{j,GABA}(t), \\ I(t) &= I_{AMPA}(t) + I_{NMDA}(t) + I_{GABA}(t) \end{aligned} \tag{2.6}$$

Because $V_i(t)$ is in the range $[V_I, V_E]$ with $V_E > V_I$, the sign of the depolarising current is negative, and positive for hyperpolarising current.

The dynamics of the fast-receptor gating variables are described by the dynamics

$$\frac{ds_{j,AMPA/GABA}(t)}{dt} = -\frac{s_{j,AMPA/GABA}(t)}{\tau_{AMPA/GABA}} + S_j(t) \tag{2.7}$$

where $S_j(t)$ is the spike train of the presynaptic neuron. The slower NMDARs have a rise process

$x_{j,NMDA}(t)$ as well, collectively described as

$$\begin{aligned}\frac{dx_{j,NMDA}(t)}{dt} &= -\frac{x_{j,NMDA}(t)}{\tau_{NMDA,rise}} + S_j(t) \\ \frac{ds_{j,NMDA}(t)}{dt} &= -\frac{s_{j,NMDA}(t)}{\tau_{NMDA,decay}} + \alpha x_{j,NMDA}(t)(1 - s_{j,NMDA}(t))\end{aligned} \tag{2.8}$$

The Wang model uses the Jahr-Stevens formula [39] to describe the dependence of the NMDA-conductance on the membrane potential

$$g_{NMDA}(V_i) = \frac{g_{NMDA}}{1 + \gamma_{JS} \exp(-\beta_{JS} V_i)}. \tag{2.9}$$

2.2 Synaptic Plasticity

The strength w_{ij} of a synapse from neuron j to neuron i can be measured as the height at the peak of the Postsynaptic Potential (PSP) or Postsynaptic Current (PSC) following a presynaptic AP or as the slope of the PSC [16]. For our purposes, and as mentioned above, the synaptic strengths w_{ij} can be seen as multipliers which scale the conductances g_{rec} of the various ion channels.

The synaptic strength w_{ij} can be decoupled into the product of the amount of neurotransmitter released into the synaptic cleft (itself a product of the release probability and the available neurotransmitter) and the density of available receptors on the postsynaptic membrane [45, 46, 47]; these variables might be treated as dynamic on a short timescale with an Short-Term Plasticity (STP) model [16, 41]. However, in this thesis w_{ij} is treated as a single isolated variable.

Most of these models consider excitatory plasticity, but there also exists literature on inhibitory plasticity models which seems to perform the distinct functions such as maintaining a balance of excitation and inhibition and thus keep the population of neurons in the asynchronous firing regime [48, 49, 50] (discussed below, in Section 2.2.4). In this thesis I will only consider plasticity of synapses with excitatory presynaptic neurons.

2.2.1 Phenomenological Models

A simple LIF model only attempts to capture the membrane potentials and the spike times of the neurons. As such, any model of plasticity built upon it can only be phenomenological, depending on the variables present in the underlying neuron model. A review of such feasible rules can be found in [16]; below I consider the relevant points.

To classify plasticity rules, it is helpful to consider the length of the period over which the changes will persist. STP rules model changes which last for under a second, and are putatively

driven by changes in the release probability and available neurostransmitter at the axon terminal [16, 41]. Long-Term Potentiation (LTP) and Long-Term Depression (LTD) describe longer lasting increases and decreases (respectively) of the synaptic strength, which can persist for more than an hour. Finally there is late-phase plasticity which occurs on much longer timescales; models such as the TagTriC model [51] or a bistability model [16] exist for this, but these timescales will not be considered here.

LTP and LTD are usually combined into a single framework, where depending on the variations within induction procedure the the same plasticity rule yields either one or the other. The induction procedures compatible with an LIF model depend on time-averaged firing rates (or low-pass filtered spike trains), which may be called rate-based learning rules, or on the precise timing of the spikes, called Spike-Time-Dependent Plasticity (STDP). These learning rules may also depend in some way on the membrane potential, such as in the Clopath model [52, 53] but more biological models such as the Shouval model [15] (which depends on calcium ion concentrations) are incompatible.

Another way to characterise plasticity rules is by whether the synaptic strengths are discrete or continuous [16]. In the framework discussed here, I only model synaptic strengths as continuous; this is justified as I will use a point-neuron model which for simplicity admits one synapse between pairs of neurons. This single synapse captures the sum of individual discrete synapses which might be modeled independently in a multi-compartment neuron model.

Finally, in a number of ways the plasticity rule can be probabilistic or deterministic. Probabilistic rules for a continuous-valued synaptic strength variable may be achieved by converting the synaptic strength dynamics $\frac{dw}{dt}$ into in Itô process, while for a discrete learning rule one can use a Markov chain over the discretely many states.[1]

2.2.2 The Volterra Series

A Volterra series expansion is an infinite sum over multidimensional convolution integrals. It can be thought of as a Taylor series which captures memory effects through the convolutions. Here I give a brief conceptual introduction to them.

Given a sufficiently nice function f taking as input a vector $\mathbf{x}_t$ indexed by time lags $\mathbf{x}_t =$

[1] For an example using a Markov chain, one can consider [24] while the Itô process is a natural extension of the drift-diffusion dynamics considered in [54] to determine the steady-state of the membrane potential distribution.

$(x_t, x_{t-\tau}, \ldots, x_{t-(n-1)\tau})^\top$ where $t = m\tau$ for some m, we can expand $f(\mathbf{x}_t)$ around $\mathbf{x}_0$ as

$$\begin{aligned}
f(\mathbf{x}_t) = f_0 &+ f_1^1(0)(x_t - x_0) \\
&+ f_1^1(0)(x_{t-\tau} - x_{-\tau}) + \cdots + f_1^1(0)(x_{t-(n-1)\tau} - x_{-(n-1)\tau}) \\
&+ f_2^{1,1}(0,0)(x_t - x_0)^2 + \cdots + f_2^{1,n}(0,0)(x_t - x_0)(x_{t-(n-1)\tau} - x_{-(n-1)\tau}) \\
&\;\;\vdots \\
&+ f_2^{j,j}(0,0)(x_{t-j\tau} - x_{-j\tau})^2 + \cdots + f_2^{j,n}(0,0)(x_{t-j\tau} - x_{-j\tau})(x_{t-(n-1)\tau} - x_{-(n-1)\tau}) \\
&\;\;\vdots \\
&+ f_2^{n,n}(0,0)(x_{t-(n-1)\tau} - x_{-(n-1)\tau})^2 \\
&+ f_3^{1,1,1}(0,0,0)(x_t - x_0)^3 + \ldots
\end{aligned}$$

In this expansion the coefficients are all functions of vectors of zero, or in another word, constants. In a Taylor expansion they would be determined by the partial derivatives of f at $\mathbf{x}_0$, but for our purposes all that matters is that such a representation exists. The zeroes are included for illustrative purposes. Rewriting, or rather reindexing, $f_1^j(0)$ as $f_1(j\tau)$, and writing x_t as $x(t)$, we can simplify this to

$$\begin{aligned}
f(\mathbf{x}_t) = f_0 &+ \sum_{k=0}^{n-1} f_1(k\tau)(x(t-k\tau) - x(-k\tau)) \\
&+ \sum_{j,k=0}^{n-1} f_2(j\tau, k\tau)(x(t-k\tau) - x(-k\tau))(x(t-j\tau) - x(-j\tau)) + \ldots
\end{aligned}$$

Now, since t is a multiple of τ we can rearrange and group terms that are multiplied by the same $x(t)$'s, introducing new coefficients $\overline{f}_n$ to catch the grouped coefficients, yielding

$$\begin{aligned}
f(\mathbf{x}_t) = f_0 &+ \sum_{k=0}^{n} \overline{f}_1(k\tau)x(t-k\tau) \\
&+ \sum_{j,k=0}^{n} \overline{f}_2(j\tau, k\tau)x(t-k\tau)x(t-j\tau) + \ldots
\end{aligned}$$

Taking the limit as x becomes a function over $\mathbb{R}$ we get the Volterra expansion

$$\begin{aligned}
f(x) = f_0 &+ \int_0^\infty \overline{f}_1(s)x(t-s)ds \\
&+ \int_0^\infty \int_0^\infty \overline{f}_2(s,s')x(t-s)x(t-s')dsds' + \ldots
\end{aligned}$$

$\overline{f}_k$ is called the k-th order Volterra kernel. This can be generalised straight-forwardly to a functional over multiple functions, such as used below.

2.2.3 Spike-Time-Dependent Plasticity

A Volterra series expansion is used on the synaptic weight dynamics in [22, 23], whereby they find the simplest functionals $\mathcal{F}$ and $\mathcal{G}$ which satisfy the experimental data of plasticity induction with various protocols. This also provides an effective general starting point for discussion of STDP rules. In their formalism, one can describe the dynamics of the synaptic strength as

$$\frac{dw_{ij}}{dt} = X(t)\mathcal{F}(X,Y) + Y(t)\mathcal{G}(X,Y) \tag{2.10}$$

where $X(t) = \sum \delta(t - t_j^f)$ is the presynaptic spike train, the sum over Dirac functions centered at the presynaptic spike times t_j^f, and similarly $Y(t) = \sum \delta(t - t_i^f)$ is the postsynaptic spike train.

The functionals $\mathcal{F}$ and $\mathcal{G}$ can be expanded into Volterra series as

$$\begin{aligned}\mathcal{F}(X,Y) = \mathcal{F}_0^X &+ \int_0^\infty \mathcal{F}_1^{X,X}(s)X(t-s)ds + \int_0^\infty \mathcal{F}_1^{X,Y}(s)Y(t-s)ds \\ &+ \int_0^\infty \int_0^\infty \mathcal{F}_2^{X,X,X}(s,s')X(t-s)X(t-s')dsds' \\ &+ \int_0^\infty \int_0^\infty \mathcal{F}_2^{X,X,Y}(s,s')X(t-s)Y(t-s')dsds' \\ &+ \int_0^\infty \int_0^\infty \mathcal{F}_2^{X,Y,Y}(s,s')Y(t-s)Y(t-s')dsds' + \dots\end{aligned} \tag{2.11}$$

and similarly

$$\mathcal{G}(X,Y) = \mathcal{G}_0^Y + \int_0^\infty \mathcal{G}_1^{X,Y}(s)X(t-s)ds + \int_0^\infty \mathcal{G}_1^{Y,Y}(s)Y(t-s)ds + \dots \tag{2.12}$$

The product of the functional outputs with the spike trains in (2.10) implies that STDP rules implement changes at the times of the pre- and postsynaptic spikes.

Classic STDP

Early STDP rules considered pairs of spikes where the magnitude of the LTP or LTD depended on the time between the spikes in the pair being considered. Typically this amplitude would decay exponentially (see Figure 2.1) leading to the STDP rule

$$\frac{dw_{ij}}{dt} = X(t)\int_0^\infty \mathcal{F}_1^{X,Y}(s)Y(t-s)ds + Y(t)\int_0^\infty \mathcal{G}_1^{X,Y}(s)X(t-s)ds \tag{2.13}$$

where

$$\begin{aligned}\mathcal{F}_1^{X,Y}(s) &= -A_- \exp\left(\frac{-s}{\tau_-}\right) \\ &=: F^{X,Y}\exp\left(\frac{-s}{\tau_{F,X,Y,0}}\right)\end{aligned} \tag{2.14}$$

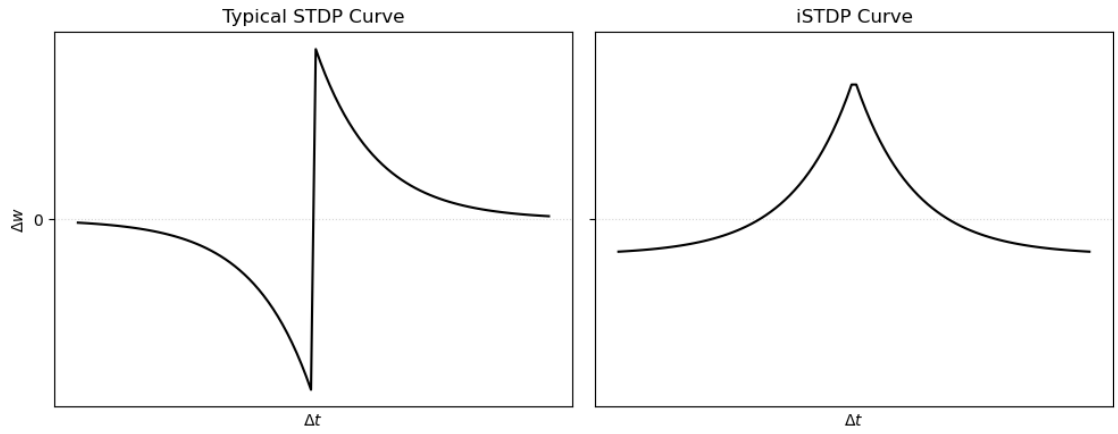


Figure 2.1: Stereotyped STDP Windows. The change in synaptic strength Δw can be approximated as a function of the difference Δt between postsynaptic spike time and presynaptic spike time for pair-based STDP rules. On the left, excitatory STDP windows typically induce pre-before-post LTP and post-before-pre LTD, while on the right the iSTDP of [48] implements LTP for coincident firing and LTD for APs spaced further apart.

and

$$\begin{aligned} \mathcal{G}_1^{X,Y}(s) &= A_+ \exp\left(\frac{-s}{\tau_+}\right) \\ &=: G^{X,Y} \exp\left(\frac{-s}{\tau_{G,X,Y,0}}\right) \end{aligned} \tag{2.15}$$

with $A_\pm \geq 0$ being the peak allowed depression or potentiation following a pre- or postsynaptic AP, respectively. These simple pair-based STDP rules with exponential decay kernels are fully characterised by $A_\pm$ and the decay times $\tau_\pm$. However, to generalise to the inclusion of other Volterra kernels, more notation is needed. I have introduced in the second lines of (2.14) and (2.15) the notation I will use for this, where the non-calligraphic F and G denote constants and $\tau_{F,X,Y,0}$ and $\tau_{G,X,Y,0}$ are the decay times. In full, and without loss of generality, we can write the k-th order exponential decay kernel for presynaptic spikes as

$$\begin{aligned} \mathcal{F}_k^{X,X,\ldots,X,Y,\ldots,Y}(s, s', \ldots, s^{(k-1)}) = F^{X,X,\ldots,X,Y,\ldots,Y} &\times \exp\left(\frac{-s}{\tau_{F,X,\ldots,X,Y,\ldots,Y,0}}\right) \\ &\times \cdots \times \exp\left(\frac{-s^{(k-1)}}{\tau_{F,X,\ldots,X,Y,\ldots,Y,k-1}}\right). \end{aligned}$$

It is notationally cumbersome yet necessary to keep track of all these decay times and coefficients.

STDP rules can be formulated as either A-A or Nearest Neighbours (N-N) [16] (see Figure 2.2). In the A-A formulation, all pairs of pre- and postsynaptic spikes are summed over, as is captured by the convolution with the spike trains in equation (2.13). However, in the alternative N-N, a presynaptic (postsynaptic) AP only induces a change in the synaptic strength determined by the difference in time from the *nearest* postsynaptic (presynaptic) APs. Whether every spike produces a change, or only every presynaptic spike, or only the presynaptic (postsynaptic) spikes which are

closest to some postsynaptic (presynaptic) spike, will determine slightly different learning rules.

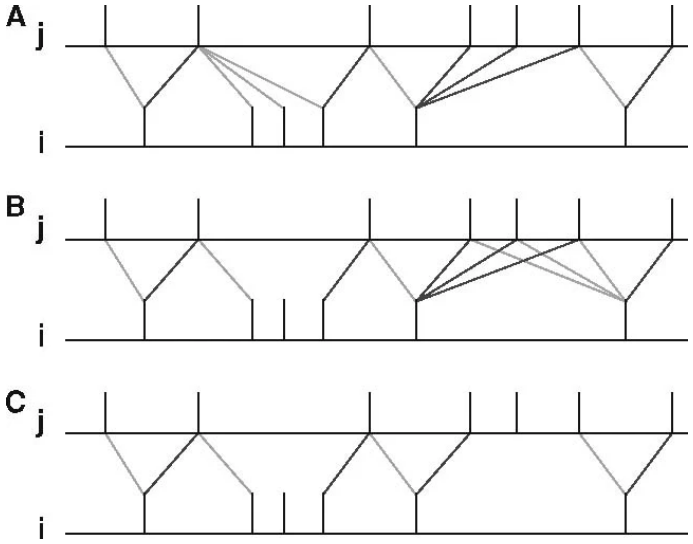


Figure 2.2: Various Nearest Neighbours spike pairing schemes. In each row, presynaptic spike trains are shown above and postsynaptic spike trains are shown below. Lines between the spike trains indicate the pairs which will be considered in the pairing scheme. Changes from dark grey lines mark pairs inducing changes at the presynaptic spike time (and induce depression in standard Hebbian STDP), while light grey lines mark pairs inducing changes at the postsynaptic spike time (inducing potentiation in in standard Hebbian STDP). In A the symmetric scheme is shown where each presynaptic spike is paired with each its nearest prior postsynaptic spike, and each postsynaptic spike is paired with its nearest prior presynaptic spike; this is the scheme used in equation (2.16). In B the presynaptic centered scheme is shown where each presynaptic spike is paired with its nearest earlier and later postsynaptic spikes; this is the scheme used in [55] and discussed in regards to BCM theory. In C the reduced symmetric scheme is shown, similar to as in A but with each spike included in at most one pair. Image taken from [16].

We can also derive a formulation for the N-N STDP with the symmetric scheme (see Figure 2.2) as follows. We denote the most recent postsynaptic AP time prior to presynaptic spike time t_j^f with $f_Y\left(t_j^f\right) = \max_t t \in \left\{t_i^f | t_i^f \leq t_j^f\right\}$, and similarly use f_X to find the last presynaptic spike time. Then, using the same kernels, we can write the classical STDP rule with N-N pairs as [22]:

$$\frac{dw_{ij}}{dt} = X(t)\,\mathcal{F}_1^{X,Y}(t - f_Y(t)) + Y(t)\,\mathcal{G}_1^{X,Y}(t - f_X(t)) \tag{2.16}$$

As in [56, 57], STDP rules often include the linear terms or zero-th order Volterra kernels $\mathcal{F}_0^X$ and $\mathcal{G}_0^X$ and possibly an extra constant homeostatic decay term C [58] yielding

$$\frac{dw_{ij}}{dt} = X(t)\left[\mathcal{F}_0^X + \int_0^\infty \mathcal{F}_1^{X,Y}(s)Y(t-s)ds\right] + Y(t)\left[\mathcal{G}_0^Y + \int_0^\infty \mathcal{G}_1^{X,Y}(s)X(t-s)ds\right] + C \tag{2.17}$$

The homeostatic decay term C can be made to be weight-dependent and/or dependent on the postsynaptic firing rate. It will be discussed below in Section 2.2.4.

At this point various questions can be asked about the learning rule, such as under what conditions the output firing rate might tend to a stable fixed point [58] or even whether the weights themselves will tend to a stable fixed point or unimodal distribution [54, 16]. For the former, inequalities can be found such as the necessary (but not sufficient) condition that if linear terms $\mathcal{F}_0^X, \mathcal{G}_0^Y$ are absent then the integral over the learning window

$$W(\Delta t) = \begin{cases} \mathcal{F}_1^{X,Y}(|\Delta t|) & \text{if} \quad \Delta t \leq 0 \\ \mathcal{G}_1^{X,Y}(\Delta t) & \text{otherwise} \end{cases} \tag{2.18}$$

is negative i.e. $\int W(\Delta t) d\Delta t < 0$, whereas if there are linear terms then $\mathcal{G}_0^Y$ needs to be sufficiently negative [58]. For the latter question, a general result is that the synaptic strengths will tend to saturate at their upper and lower bounds, or more precisely the steady-state for the Fokker-Planck dynamics of synaptic strength distribution will be bimodal near these bounds unless the plasticity dynamics depend on the weights themselves [54, 59, 16]. However, in observed populations of neurons, the synaptic strengths tend to be unimodal and generally close to lognormally distributed [60]. Suffice to say, the zero-th order kernels cannot be neglected and the kernels need to depend on the synaptic strengths, as I will discuss below in the Weight Dependence subsection.

Postsynaptic Rate Modulation and the Triplet Rules

Another way to modulate the postsynaptic firing rates is to choose a learning rule which satisfies the criteria of BCM theory.[2] BCM theory sets out several criteria such that when satisfied by a rate-based learning rule certain predictions can be made, such as that the dynamics of the model will stabilise. Such BCM models can account for a range of experimentally observed features such as the development of receptive fields, ocular dominance and synaptic scaling [62]. The criteria are:

1. The change in synaptic strength should be linear in the presynaptic firing rate, which we can denote ν_j.

2. For low postsynaptic firing rates ν_i below a threshold θ_{thr}, the synaptic change should be depressing, while for high postsynaptic firing rates the synaptic change should be potentiating (see Figure 2.3).

3. The threshold itself, θ_{thr}, should be a superlinear function of the time-averaged postsynaptic firing rate. It is this adaptive threshold that ensures the dynamics stabilise.

Although the theory is developed for rate-based plasticity rules, STDP rules can also satisfy these criteria *on average* and thus account for the same experimental observations.

[2] BCM theory is named after the researchers Bienenstock, Cooper, and Munro, authors of [61].

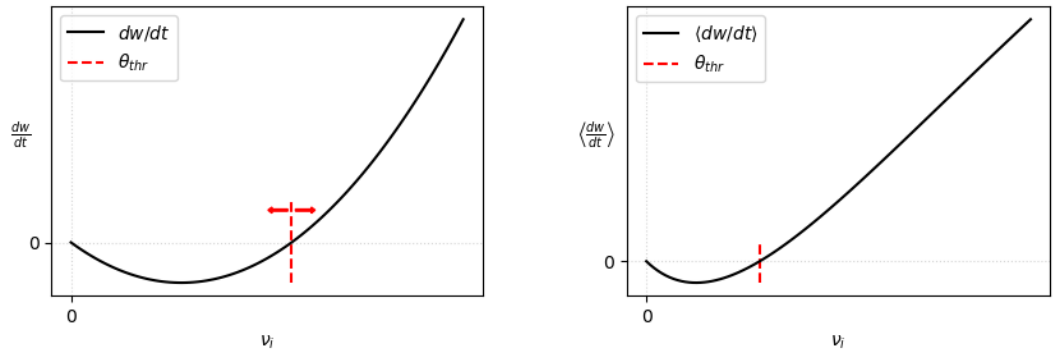
Attempts to reconcile BCM theory with STDP rules seem to begin with [55], where it was shown that an N-N STDP rule much like (2.16) with mild assumptions, such as Poisson-like activity of the pre- and postsynaptic firing, yields a learning rule which, when averaged across spike pairs, meets the criteria of BCM theory.[3] The synaptic dynamics are given by

$$\left\langle \frac{dw_{ij}}{dt} \right\rangle = \nu_i \nu_j \left(\frac{F^{X,Y}}{\nu_i + 1/\tau_{F,X,Y,0}} + \frac{G^{X,Y}}{\nu_i + 1/\tau_{G,X,Y,0}} \right) \tag{2.19}$$

This matches some of the requirements of BCM theory, as can be seen in Figure 2.3. However, an extra mechanism needs to be added to allow for the BCM threshold, determined as

$$\theta_{thr} = -\frac{G^{X,Y}/\tau_{F,X,Y,0} + F^{X,Y}/\tau_{G,X,Y,0}}{F^{X,Y} + G^{X,Y}},$$

to be adaptive; in [55] it is suggested that $\tau_{G,X,Y,0}$ might vary depending on the state of the NMDARs.



(a) The BCM Rule.

(b) Classical N-N STDP.

Figure 2.3: The threshold θ_{thr} changes as a superlinear function of the postsynaptic firing rate ν_i, while the magnitude of $\frac{dw}{dt}$ depends linearly on the presynaptic rate ν_j and non-linearly on the postsynaptic rate. In Figure 2.3a the arrows indicate that the threshold is adaptive. In Figure 2.3b the average synaptic strength follows dynamics qualitatively similar to the BCM rule. Figure 2.3a adapted from www.scholarpedia.org/article/BCM_theory. Figure 2.3b computed using parameters given in [55] and formula (2.19).

Another way to satisfy the BCM criteria is to consider higher-order Volterra kernels and explicitly add the dependence on the postsynaptic firing rate. Triplet STDP rules selectively consider pairs and triplets of spikes: in [22] only $\mathcal{F}_1^{X,Y}$ and $\mathcal{G}_2^{X,Y,Y}$ are considered, while in [23] the full triplet model with both first order kernels are considered alongside $\mathcal{F}_2^{X,X,Y}$ and $\mathcal{G}_2^{X,Y,Y}$, and in [63] both first order kernels are considered alongside only $\mathcal{G}_2^{X,Y,Y}$. In all cases the kernels are of the form of exponential decay, and if the depressive terms kernels (in these models, $\mathcal{F}_1^{X,Y}$ and $\mathcal{F}_2^{X,X,Y}$)

[3] It should be noted that the rule in [55] is presynaptic-centered, as explained in [16] and shown in Figure 2.2. That is, each presynaptic spike is paired with the nearest postsynpaptic spikes before and after it, but some postsynaptic spikes are not paired at all unless they are nearer to some presynaptic spike - either occurring before or after - than any other postsynaptic spike.

are given a superlinear dependence on a low-pass filtered postsynaptic firing θ where

$$\tau_\theta \frac{d\theta}{dt} = -\theta + Y(t) \tag{2.20}$$

then the BCM theory criteria are recovered. Here θ should not be confused with the threshold θ_{thr} in BCM theory. In fact, one may use the relationship

$$\theta_{thr} = \left(\frac{\theta}{\nu_0}\right)^p$$

where ν_0 is a baseline postsynaptic firing rate provided $p > 1$ to implement the superlinearity dependence.

Ultimately in [22] they use functionals of the form (where I've absorbed the baseline firing rate into the coefficient)

$$\begin{aligned} \mathcal{F}_1^{X,Y}(s;\theta) &= \theta^2 F^{X,Y} \exp\left(\frac{-s}{\tau_{F,X,Y,0}}\right), \\ \mathcal{G}_1^{X,Y,Y}(s,s') &= G^{X,Y,Y} \exp\left(\frac{-s}{\tau_{G,X,Y,Y,0}}\right) \exp\left(\frac{-s'}{\tau_{G,X,Y,Y,1}}\right), \end{aligned} \tag{2.21}$$

and similarly in [23] and [63].

For A-A interactions, this yields trial-averaged plasticity rules of the form

$$\left\langle \frac{dw_{ij}}{dt} \right\rangle = \theta^2 F^{X,Y} \tau_{F,X,Y,0} \nu_j \nu_i + G^{X,Y,Y} \tau_{G,X,Y,Y,0} \tau_{G,X,Y,Y,1} \nu_j \nu_i^2 \tag{2.22}$$

and for N-N interactions, rules of the form

$$\left\langle \frac{dw_{ij}}{dt} \right\rangle = \frac{\theta^2 F^{X,Y} \nu_j \nu_i}{\nu_i + 1/\tau_{F,X,Y,0}} + \frac{G^{X,Y,Y} \nu_j \nu_i^2}{(\nu_j + 1/\tau_{G,X,Y,Y,0})(\nu_i + 1/\tau_{G,X,Y,Y,1})} \tag{2.23}$$

Aside from the BCM results, the triplet rules also have various other nice features. They can generalise the BCM results to higher order statistics [63] and can account for yet more experimental protocols [23]. As an example, consider the LTP induction protocol of inducing a presynaptic spike followed by a postsynaptic spike some Δt_1 time later, then repeating this procedure after a time $\Delta t_2 > \Delta t_1$. As $\Delta t_2 \to \Delta t_1$, the classic STDP protocol (2.13) will predict *less* potentiation as the time between successive post-pre pairs decreases. However, experimentally *more* potentiation is observed, as is predicted by these triplet models.

The observed dependence on triplets of spikes might be an epiphenomenon arising from the dependence on postsynaptic membrane potential fluctuations. This dependence is directly modelled in the Clopath model [52], and other voltage-dependent plasticity rules. If the membrane potential fluctuations are induced primarily by BPAPs then the Clopath model and the triplet model of [22] become equivalent [38]. This suggests that little is lost when considering a biophysically inspired

SNN with plasticity dependent only on spike times.

Finally, the triplet rules (and voltage-dependent rules) allow for more complex network topologies to arise [53]. Indeed, the standard STDP rule all but forbids strong bidirectional connections due to the learning window W being qualitatively antisymmetric around $\Delta t = 0$: that which is a pre-before-post pair of spikes for the synapse ij from neuron j to neuron i is a post-before-pre pair for the synapse ji. This can however be circumvented somewhat by incorporating axonal and dendritic delays [64, 16].

Weight Dependence

There is direct biological evidence that the changes in synaptic strengths over a trial depend on the initial synaptic strengths [65, 16], as well as indirect evidence for this, namely that unimodality of the synaptic strength distribution will not be achieved with a pair-based plasticity rule if increments in synaptic strength do not depend on the present value itself [54, 59, 16]. This is fairly intuitive: as synaptic strength increases, the expected time to the next postsynaptic spike time following a presynaptic spike decreases, which increases the value in the LTP side of the learning window and leads to further potentiation. To offset this, the integral of the LTD side of the window needs to be greater (in magnitude) than that of the LTP side, but if it is too large it leads to runaway depression.

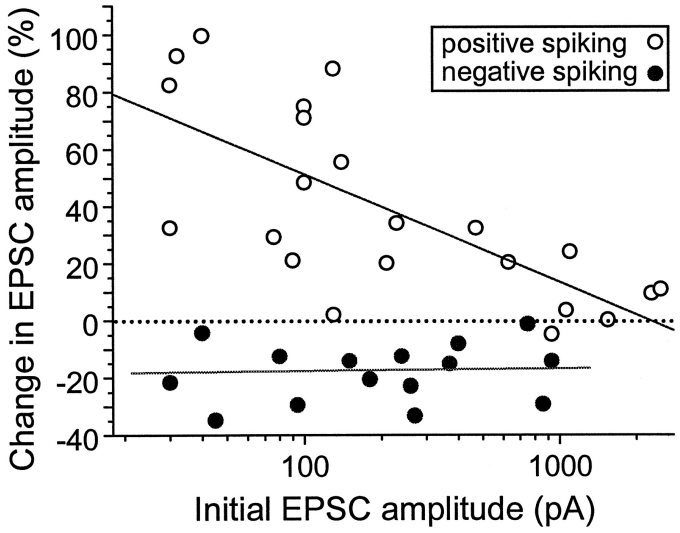


Figure 2.4: Percentage change of synaptic strength, measured in EPSC amplitude. Empty circles show changes in synapses exposed to presynaptic stimulation with positively correlated postsynaptic spiking, leading to LTP. Filled circles show the same, but with negatively correlated postsynaptic spiking, leading to LTD. Data is plotted as a function of mean initial EPSC amplitude. The straight line fitted for LTD suggests that the absolute change in EPSC amplitude depends multiplicatively on initial EPSC amplitude. Image taken from [65].

A multiplicative weight dependent update of the form

$$\mathcal{F}_1^{X,Y}(s, w_{ij}) = w_{ij}\mathcal{F}_1^{X,Y}(s)$$

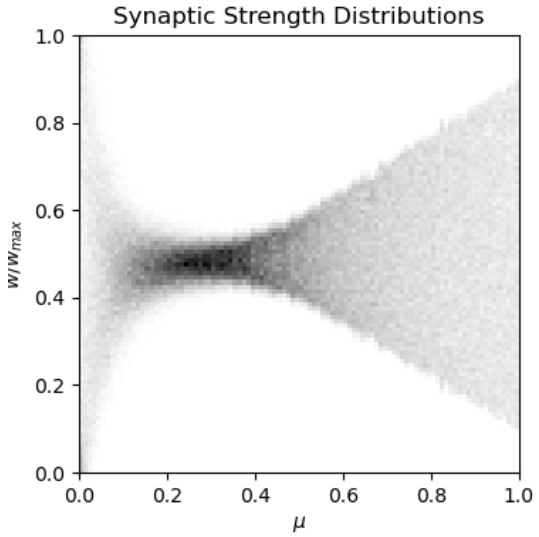


Figure 2.5: Empirical probability densities of synaptic strengths after 100 seconds of simulation time. Probability density is plotted with darker shades corresponding to high probabilities. This is a reconstruction of the plot from [16] with a wider range of values for μ, using STDP and LIF parameters from brian2.readthedocs.io/en/2.0rc/examples/synapses.STDP.html.

can be fitted for the LTD data of [65], while the LTP data is better fitted with a power law update with power μ in the form

$$\mathcal{G}_1^{X,Y}(s, w_{ij}) = w_{ij}^{\mu} \mathcal{G}_1^{X,Y}(s)$$

An alternative form of weight dependence is also suggested in [16], to implement soft upper and lower bounds (w_{max} and 0, respectively) using rules of the form

$$\begin{aligned} \mathcal{F}_1^{X,Y}(s, w_{ij}) &= w_{ij}^{\mu}\, \mathcal{F}_1^{X,Y}(s), \\ \mathcal{G}_1^{X,Y}(s, w_{ij}) &= (w_{max} - w_{ij})^{\mu}\, \mathcal{G}_1^{X,Y}(s) \end{aligned} \tag{2.24}$$

with $\mu \in [0, 1]$. Classical STDP with kernels of this form in (2.24) yield unimodal distributions for a wide range of values for μ, as can be seen in Figure 2.5.

2.2.4 Rate-Based Plasticity

As has been seen in the Postsynaptic Rate Modulation and the Triplet Rules subsection, and has been discussed in [56, 57, 66, 58, 55, 23, 22, 16, 38], if one assumes that the pre- and postsynaptic spike trains are generated by a Poisson process - where the latter may be an inhomogeneous Poisson process driven by the former - then one can derive various analytical results regarding the trial-averaged behaviour and synapse-average behaviour (assuming neurons are homogeneous and their activity is effectively uncorrelated, which may be nearly true in the limit of many neurons) of the network, such as querying the presence of postsynaptic rate modulation [58] or determining the fixed point of the synaptic weights dynamics [16]. This is usually done for simple neuron models,

as the correlation function between pre- and postsynaptic spike times can be difficult to determine analytically for complex models [56]. However, this becomes more difficult as higher-order Volterra kernels are included in the rule, as the correlation function itself now depends on multiple differences in time between different spikes. Nonetheless, in the limit of many weak synapses this correlation all but vanishes and the approximation that the pre- and postsynaptic spike trains are uncorrelated becomes more feasible.

This approximation of inhomogeneous Poisson spike trains allows us to consider plasticity rules of the form [22]

$$\begin{aligned}
\left\langle \frac{dw_{ij}}{dt} \right\rangle &= \langle X(t)\mathcal{F}(X,Y)\rangle + \langle Y(t)\mathcal{G}(X,Y)\rangle \\
&= \left\langle X(t)\left[\mathcal{F}_0^X + \int_0^\infty \mathcal{F}_1^{X,X}(s)X(t-s)ds + \dots\right]\right\rangle \\
&\quad + \left\langle Y(t)\left[\mathcal{G}_0^Y + \int_0^\infty \mathcal{G}_1^{Y,X}(s)X(t-s)ds + \dots\right]\right\rangle \\
&= \langle X(t)\rangle\left[\mathcal{F}_0^X + \left\langle \int_0^\infty \mathcal{F}_1^{X,X}(s)X(t-s)ds\right\rangle + \dots\right] \\
&\quad + \langle Y(t)\rangle\left[\mathcal{G}_0^Y + \left\langle \int_0^\infty \mathcal{G}_1^{X,Y}(s)X(t-s)ds\right\rangle + \dots\right] \\
&= \langle X(t)\rangle\left[\mathcal{F}_0^X + \langle X(t)\rangle F^{X,X}\tau_{F,X,X,0} + \dots\right] \\
&\quad + \langle Y(t)\rangle\left[\mathcal{G}_0^Y + \langle X(t)\rangle G^{X,Y}\tau_{G,X,Y,0} + \dots\right]
\end{aligned} \tag{2.25}$$

where the first step is simply using the Volterra expansion, the second step relies on independence of the spike trains X and Y, and the final step uses exponential decay kernels for $\mathcal{F}$ and $\mathcal{G}$. Considering that the neurons are assumed to be homogeneous (at least within the same populations) and nearly independent, we get that the average firing rate ν_k of a population of neurons (where k denotes the population index) is nothing but a trial average estimate of the firing rates of the individual neurons within the population [38]. So if we consider the presynaptic neuron j to be in population k' and the postsynaptic neuron i to be in population k, this reduction (2.25) gives us the population averaged dynamics

$$\left\langle \frac{dw_{ij}}{dt} \right\rangle = \nu_{k'}\left[\mathcal{F}_0^X + \nu_{k'}F^{X,X}\tau_{F,X,X,0} + \dots\right] + \nu_k\left[\mathcal{G}_0^Y + \nu_{k'}G^{X,Y}\tau_{G,X,Y,0} + \dots\right] \tag{2.26}$$

whence we get rules of the form (2.22) [38]. To consider the role of only admitting N-N pairs, the procedure is more involved but the result is a rule of the form we saw in (2.23) [22, 23].

How realistic is the assumption that the spike trains are generated by inhomogeneous Poisson processes? That depends on the input strength. Broadly speaking, a population of spiking neurons may be in one of four activity regimes [67]:

1. *asynchronous regular* activity, where the individual neurons fire regularly but the population rate is roughly constant;

2. *synchronous regular* activity, where the neurons fire regularly and the population firing rate oscillates;

3. *synchronous irregular* activity, where the individual neurons fire irregularly but the population rate oscillates nonetheless;

4. *asynchronous irregular* activity, where the population rate is roughly constant and the neurons fire irregularly.

This last regime may be the norm, encouraged by inhibitory plasticity [48, 49, 50]. Which regime a population of neurons finds itself in depends on the strength of the input current and the noise in the input to the neurons: under strong inputs, where the mean input is strong enough to induce spiking, the neurons adopt regular spiking (see Figure 2.6 for a schematic) which, in the absence of noise in a current-based LIF model, is guaranteed to induce synchronisation [68], but will generally lead to synchronisation for a broader class of models. If excitatory and inhibitory inputs to the neuron are balanced, such that the mean input is not strong enough to induce spiking behaviour but the *fluctuations* in the input can allow the membrane potential to rise sufficiently to induce an AP, then the neuron is said to be in the balanced regime and the spike train statistics appear to be Poisson-like, with an exponential Interspike Interval (ISI) distribution. Indeed, these asynchronous and irregular population dynamics might be fingerprints of chaos in sufficiently large neural networks, although it is common to treat the unpredictable inputs as random noise in smaller models [67]. In short, one might approximate the neurons spike trains in the asynchronous irregular regime as being generated by independent inhomegeneous Poisson processes with rates given by the population firing rate, reducing the task of modeling the population to that of modeling only the population firing rate.

Historically, before observations of dependence of plasticity on precise spike times, one would start with a rate-based learning rule. Many such rate-based rules can be shown to be special cases of the averaged rules considered above. We have seen that this is the case for the BCM rule. Another such example is Oja's rule which arises from considerations of stable learning of the covariance of the input stimuli by implementing weight decay [41, 38].

Weight Decay and Heterosynaptic Plasticity

Heterosynaptic plasticity refers to the process whereby activity at one synapse tends to influence the plasticity of other synapses on the shared postsynaptic neuron. Homosynaptic plasticity, by contrast, does not seem to provide sufficient competition between synapses to drive effective learning, nor to discourage runaway growth of weights [69]. In the absence of a weight-dependent update rule such as discussed in Section 2.2.3, heterosynaptic plasticity provides a way to stabalise weights and consequently postsynaptic activity levels.

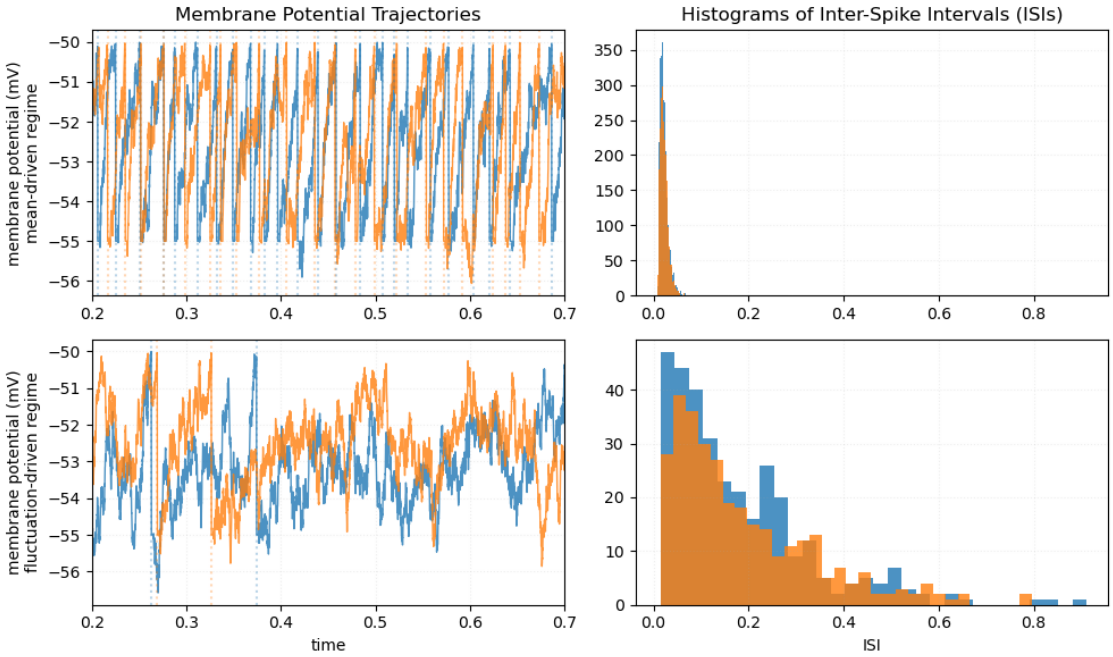


Figure 2.6: Comparison of firing regimes. In the first row, the mean-driven regime, the two LIF neurons are driven by the mean of a noisy input, leading to regular firing and a narrow ISI distribution. In the second row, the balanced regime or fluctation-driven regime, the synaptic inputs (here modeled by a white noise process) are balanced so that only fluctuations in the noise drive the neurons to fire. This leads to an almost exponential ISI distribution. In the left column, trajectories of membrane potentials of two neurons over 500ms are shown. Spike times are indicated by dotted lines. In the right column, ISI distributions are shown for the same trajectories over a longer time.

While it seems that heterosynaptic mechanisms might be local to regions along a dendritic branch [69], in point-neuron models such locality cannot be implemented.

Heterosynaptic plasticity may also play a role in long-term consolidation of memories and homeostatic processes, but such homeostatic processes seem to happen on a timescale much slower than what is required for the stability of models, suggesting a dual mechanism of short term heterosynaptic plasticity - called a Rapid Compensatory Process (RCP) - and long-term homeostatic effects [70].

Heterosynaptic plasticity is primarily implemented via implementing a weight decay in one of two forms [71, 41]. In subtractive weight decay, the synaptic weights are bound by their cumulative strength, and each weight decays with the same increment. The plasticity rule obtains a term such as

$$\frac{dw_{ij}}{dt} = \cdots - c \sum_j w_{ij}^m \tag{2.27}$$

where m might be 1 or 2 and $c > 0$. If synaptic weights are allowed to be negative, then their absolute values $|w_{ij}|$ would be used. Such a learning rule is arguably non-local, however, as all the synapses need to be "aware" of the strength of the other synaptic weights which may in part be encoded presynaptically [46], and hence may not be particularly biologically feasible.

A more biologically feasible alternative form of weight decay, multiplicative weight decay, adds a term of the form

$$\frac{dw_{ij}}{dt} = \cdots - c\,\nu_i^n w_{ij}^m \tag{2.28}$$

The mechanism for heterosynaptic plasticity in this type of local weight decay can be seen by approximating the input-output function by a linear function i.e. approximating $\nu_i = \sum_j w_{ij}\nu_j$. Inserting this into (2.28) gives

$$\frac{dw_{ij}}{dt} = \cdots - c\sum_k \nu_k^n w_{ik}^{m+n}$$

Stimulating the postsynaptic neuron into firing more indirectly decreases the strength of other synapses onto the same neuron, thus implementing competition between the synapses.

Oja's Rule

Oja's rule can be seen as a special case of implementing multiplicative weight decay (2.28) [41]. It takes the form

$$\frac{dw_{ij}}{dt} = \nu_i\nu_j - c\,\nu_j^2 w_{ij} \tag{2.29}$$

When implemented in a rate model with firing dynamics of the form $\nu_i = \sum_j w_{ij}\nu_j$, this rule performs online principal component analysis on the input stimuli by driving the weight vector to align with the first principal component, thus causing the postsynaptic firing rate to be a projection of the stimuli onto that component. By adding lateral inhibition in a two layer model, Sanger's Rule (or the Generalised Hebbian Algorithm) allows the input stimuli to be projected onto the k-th principal subspace where k corresponds to the number of postsynaptic neurons [72]. Thus it may prove to be a convenient mechanism for early unsupervised learning of stimuli, although these rules were considered in feedforward linear networks and not the recurrent neural networks used in this thesis.

2.2.5 Three-Factor Learning Rules

All of the rules considered until now are unsupervised (in the machine learning nomenclature) or Hebbian [37] in that they rely on statistical relationships between stimuli to drive their plasticity. However, the class of problems which can be solved with such learning rules is restricted, and these rules say nothing of what role neuromodulator-encoded signals such as reward or novelty might play in the plasticity [73].

On the one hand, attempts have been made to experimentally determine the dependence of synaptic plasticity on neuromodulators [2], while on the other hand theoretical studies have attempted to derive descriptive or normative accounts of how synaptic plasticity should alter

with a reward signal (for descriptive accounts, see [74, 75, 76, 27]; for normative accounts, see [77, 78, 79, 80, 81]; a review can be found in [73] and further reviewed in [3]).

The plasticity rules up to here now depended only on the presynaptic and postsynaptic activity, as well as potentially on the current strength of the synapse. In brief, they have been rules of the form

$$\frac{dw_{ij}(t)}{dt} = H_2(\text{pre}, \text{post}; w_{ij}(t))$$

where pre and post capture the relevant local features (rates, membrane potentials, spike trains, etc.) and H_2 is an arbitrary plasticity rule, a functional over pre and post parameterised by $w(t)$. To incorporate the potential for a modulatory signal M, one can extend this framework to accommodate so-called three-factor (or neoHebbian) learning rules [73, 3]:

$$\frac{dw_{ij}(t)}{dt} = H_3(M, \text{pre}, \text{post}; w_{ij}(t))$$

where M describes the modulatory signal and H_3 is an arbitrary three-factor learning rule. In principle one can repeat the Volterra expansion, but the number of kernels grows combinatorially with the number of function inputs, and more so when considering that M might be vector-valued for all the relevant neuromodulators. Herein I will consider a simpler approach. I will restrict the approach to a one-dimensional reward signal $R(t)$ which is thought to be losslessly encoded in the neuromodulators such that there exists a g where $R(t) = g(M(t))$, and I consider the task of maximising this total reward $\int R dt$.

Descriptive Models

A simple approach might be to allow the reward signal to directly gate plasticity, such is in [74], yielding a rule such as

$$\frac{dw_{ij}(t)}{dt} = R\, H_2(\text{pre}, \text{post}; w_{ij}(t))$$

But this leads to the problem that the reward signal needs to be co-occurring with the neural activity that gave rise to it. In behavioural studies, this is known as the *distal reward problem*: how do rewards received at later times reinforce behaviour from earlier times? This is related to the temporal credit assignment problem of RL: determining which actions or features from a prior time were responsible for a reward or instructive signal at a later time. These can be resolved by introducing an eligibility trace, an idea taken from RL [75, 76].

Introducing an eligibility trace e_{ij} gives us a learning rule of the form

$$\begin{aligned} \tau_e \frac{de_{ij}}{dt} &= -e_{ij} + H_2(\text{pre}, \text{post}; w_{ij}) \\ \frac{dw_{ij}}{dt} &= e_{ij}\, R \end{aligned} \tag{2.30}$$

For example, if we restrict H_2 to consider only pre- and postsynaptic spike trains and a low-pass filtered postsynaptic spike train θ we get the Reward-Modulated Spike-Time-Dependent Plasticity (R-STDP) rule [75]

$$\begin{aligned} \tau_e \frac{de_{ij}}{dt} &= -e_{ij} + H_2(X, Y; \theta, w_{ij}) \\ \frac{dw_{ij}}{dt} &= e_{ij}\, R \end{aligned} \tag{2.31}$$

If we replace reward with its zero-meaned value $D = R - \langle R \rangle$ then using H_2 as the simple classical STDP allows a network to solve precise-timing biofeedback-like tasks [76].

Conversely, if R does not have zero-mean we can factor it as $R = \Delta R - \langle R \rangle$ where ΔR is the fluctuations of R around $\langle R \rangle$. Hence the dynamics become

$$\frac{dw_{ij}}{dt} = e_{ij}\, \Delta R - e_{ij}\, \langle R \rangle\,.$$

In this case simulations show that we require the STDP window to have zero-mean [73] or that it must be slightly negative as in [75] or the learning dynamics will be dominated by the Hebbian or unsupervised component $e_{ij}\, \langle R \rangle$ while the rewarded component $e_{ij} \Delta R$ is suppressed [80].

Moreover, if multiple tasks are to be learnt simultaneously, then $\langle R \rangle$ needs to be computed at a state-specific level where the state is taken simultaneously over the tasks i.e. we need to replace $\langle R \rangle$ with $\langle R | \text{state} \rangle$.[4] This way rewards received for correct or improved behaviour on one task do not reinforce behaviour on the other task. This requires an extra mechanism to estimate $\langle R | \text{state} \rangle$, known as a critic in RL theory [79, 81, 73, 82], but can lead to much faster learning as the plasticity rule does not need to rely on the averaging of covariance estimates as done in covariance-driven rules (described below).

Covariance-Driven Rules

The low-pass filter, or eligibility trace, of the first line in (2.30) can be thought of as computing a running average of H_2, which we can denote as $\overline{H}_2$,[5] which allows us to describe (2.30) in a single line [73]

$$\frac{dw_{ij}}{dt} \propto R \overline{H}_2$$

[4] State here refers to the state in a RL description i.e. the state of the Markov decision process.

[5] The reader is advised to note that the overline notation is not used for running averages throughout this thesis, only here.

which means the expected weight change across trials is given by

$$\left\langle \frac{dw_{ij}}{dt} \right\rangle \propto \left\langle R\overline{H}_2 \right\rangle$$
$$\propto \mathrm{Cov}(R, \overline{H}_2) - \langle R \rangle \left\langle \overline{H}_2 \right\rangle$$

Replacing R with $D = R - \langle R \rangle$, as in [76] - which provides neat theoretical guarantees and the ability to solve complex tasks - or choosing a plasticity rule H_2 such that $\left\langle \overline{H}_2 \right\rangle_{\text{trials}} = 0$, as is found with the policy gradients methods [73] (discussed below) implies that the plasticity will be driven by the covariance of the reward signal and filtered proposed weight changes H_2. Replacing H_2 with *any* neural activity signal N and using a learning rule of the form

$$\left\langle \frac{dw_{ij}}{dt} \right\rangle = \mathrm{Cov}(R, N) \tag{2.32}$$

yields a covariance-driven learning rule [83, 84]. Such learning rules have as a fixed-point matching law behaviour, which is often experimentally observed and which can maximise reward in stationary foraging tasks [85].

Since matching law behaviour would not make sense in the context of my tasks, I shall not review this further save to say that undermatching is often observed, which may be optimal in non-stationary environments when there are multiple timescales present, and which may arise due to separate learning mechanisms with their own distinct timescales [85].

Normative Models

In policy gradient learning [82] the agent adapts the parameters describing its policy - or distribution over action choices - directly by increasing the probability of repeating actions which led to reward. In a multi-agent environment one finds that the policy gradient update distributes over the agents; treating various neurons as independent agents in a multiagent environment, Bartlett and Baxter [86] arrive at a policy-gradient learning rule for spiking neural networks. Building on their work for infinite-horizon policy gradient learning [87], Florian in [78] extended this rule using the Spike Response Model (SRM) (of which the basic current-based LIF model is a special case) with escape noise [38] and considered the continuous time case. In the discrete time setting a policy gradients rule can also be determined directly with an LIF model extended with adaptation as in [77].

What these methods have in common is that they result in a learning rule of the form $H_{\mathrm{PG}}(\mathrm{pre}, \mathrm{post}) = H_2(\mathrm{pre}, \mathrm{post}) - \langle H_2(\mathrm{pre}, \mathrm{post}) | \mathrm{pre} \rangle$ [73]. Generally these rules have the curious property of only predicting LTP but when considering postsynaptic rate modulation, one can arrive at an LTD component as well [78]. However, it is not clear that the learning rules I will consider admit an

estimate $\langle H_2(\text{pre}, \text{post})|\text{pre}\rangle$.[6]

2.3 Decision Making

It is all well and good to consider how plasticity behaves in the presence of a reward signal, but it is likely that the reward signal comes indirectly from interactions with the environment. The full biological architecture of action selection, including the direct and indirect pathways and the anatomy of the basal ganglia, are well outside the scope of this thesis. In what follows I will consider only a neural correlate of decision making, that is the firing rate of individual neurons in some particular region of the brain - in our case, the LIP area - and how a recurrent neural circuit model of their activity can be used to simulate action choice.

Modeling the (perceptual) decision making process is more than trying to match the proportion of correct and incorrect responses. Theoreticians attempt to account for features of the decision making process, such as the time taken to make incorrect decisions, or whether and how the accuracy of decisions made is changed by adding a time delay between the cessation of the perceptual stimulus and the response time, that is, the time at which the decision is made [17]. Various different models of this process exist, as reviewed in [17]. For our purposes we will consider Drift Diffusion (DD) models, which may be linear as in [18] or nonlinear such as in [88].

Neural correlates of decision making have been observed in the mammalian brain to arise from collective dynamics of neurons [17]. Biophysically inspired models have been developed to bridge the levels of description from network and cellular mechanisms to behavioural performance. A subset of the models, called "recurrent neural circuit models" [17], are characterised by a few distinguishing features: recurrent synaptic excitation is assumed to be sufficiently strong so as to be able to generate multiple persistent states of increased activity, or attractor states, and that this reverberating excitation is instantiated by slower cellular processes leading to a slow ramping of neural activity similar to the increase in population activity observed in the LIP area when performing the RDM task (discussed below), that feedback inhibition is incorporated to instantiate competition between neurons and finally that stochastic choice behaviour arises from irregular spiking activity. This last feature we will drop, and consider that there simply needs to be intrinsic noise in the model, so that we can include rate-based models in this class.

One such model with sufficient generality, which will be used herein, is the model by Wang [20], adapted from [19] for exactly this purpose of explaining LIP data on monkeys performing the RDM task.

[6] While escape noise models can compute this expectation explicitly using the escape rate, diffusive noise as used in the reduced Wang model is much less tractable and it is not apparent what that a biological substrate for this expectation would be present.

In what follows, first I will describe the RDM task that the Wang model was adapted to account for, then I will discuss DD models of decision making in general as contrasted with recurrent neural circuit models. Finally I will discuss the Wang model itself.

2.3.1 The Random Dot Motion Task

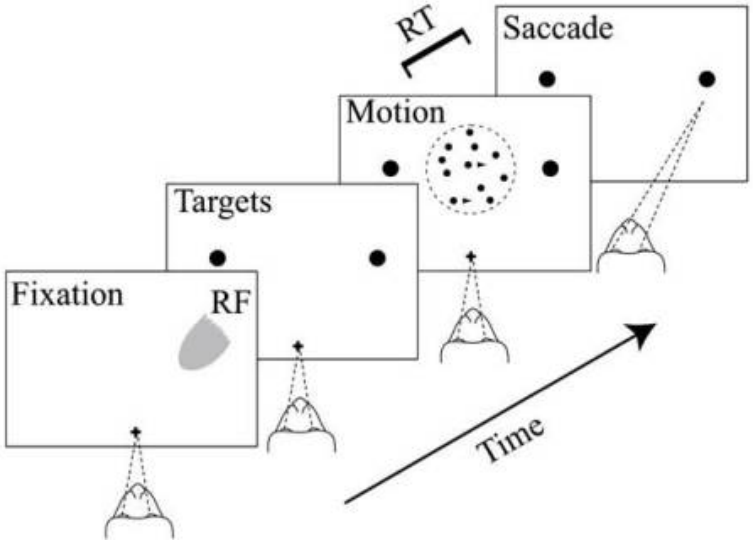


Figure 2.7: The Random Dot Motion Task. The subject is required to fixate on a point on a screen while dots are displayed moving in random directions. A fraction of dots (known as the coherence) move in the same direction. Afterwards, the subject is required to perform a saccade in the direction in which these dots moved. The task comes in two forms: either the subject is cued to make a response, or allowed to determine when to make a response on their own. In the former case one can measure performance as a function of the exposure-to-stimulus time or delay time before the cue, while in the latter one can evaluate the relationship between RT and decision accuracy. Image taken from [17].

The RDM task is the canonical task that will be used in the EAs here. It is an example of a 2-alternative forced choice perceptual decision making task. In it, the subject - such as a monkey - is taught to fixate at a point on a screen. While doing so, several moving dots are displayed on the screen. A fraction of these dots, called the coherence, will be moving in the same direction, one of two directions, while the remaining dots move about randomly. The subject is required to perform a saccade in the direction of the coherently moving dots and if successful receives a reward. In the fixed-duration version of the task, the subject must wait for a cue at which time they must perform the task. In the reaction time (RT) version, the subject can choose to perform the saccade whenever they are ready [17].

In RT tasks experimentalists can assess the relationship between reaction times and accuracy or error rate, while in the fixed-duration tasks experimentalists can alter the difficulty of the tasks by adjusting the time of exposure to the random dots stimulus. Recordings of cellular activity can be done while the task is performed, particularly of the Middle Temporal (MT) area, which responds to motion in the visual field, and the LIP area, which is implicated in decision making

[17]. What is typically observed is that neural populations in the MT area respond most strongly to the signal, increasing their firing rate in proportion to their selectivity for motion in the direction of the coherent subset of dots, while neurons in the LIP area increase their firing rate in proportion to the decision made [17]. Moreover, on RT trials once the population activity reaches a threshold the saccade is performed, while on tasks where there is a delay before the saccade cue, neural activity remains near this threshold until the cue is received. Thus the population activity of cells in the LIP area serves as a neural correlate for the decision making process.

A few other salient features of these experiments are [17]: Firstly, on error trials, the RT is usually longer than on correct trials. Secondly, even if the exposure to the stimulus is extended in fixed-duration trials, performance on the trials plateaus early and there remains a non-zero error probability. Thirdly, if a brief motion pulse is included as part of the signal, the effect of this pulse is more significant if it is provided earlier on, suggesting that earlier stimuli are more important in the decision making process than later stimuli. All of these features are recreated in recurrent neural circuit models, as well as one other: if a delay is provided on fixed-duration trials between cessation of the stimulus and the saccade cue, performance does not noticeably drop. The first three features are features which linear DD decision making models do not naturally replicate, while the last feature leaky accumulator models - viable alternatives to DD models, but not which will not be considered here - fail to replicate. Altogether, recurrent neural circuit models capture a wide range of experimental data.

2.3.2 Drift Diffusion Decision Making

In the Sequential Probability Ratio Test (SPRT) one repeatedly samples data as evidence e_t for one or another of a pair of hypotheses H_0 and H_1. One defines a decision variable X_{DV} following dynamics

$$X_{DV}(t) = X_{DV}(t-1) + \log \frac{\mathcal{P}(e_t|H_0)}{\mathcal{P}(e_t|H_1)}, \quad X_{DV}(0) = 0 \tag{2.33}$$

and continues updating X_{DV} until it crosses one or another threshold. The threshold crossed determines the choice of the hypothesis. If each e_t is independently sampled, then the increments/decrements to X_{DV} are should be independent of its current value. As such, the SPRT is optimal [18].

In a DD model for decision making, the latent decision variable X_{DV} is modeled as following a DD process with drift U and diffusion D such that

$$dX_{DV}(t) = U(X_{DV}(t))dt + DdW_t, \quad X_{DV}(0) = 0 \tag{2.34}$$

where W_t is a 1-dimensional Wiener process. In the case of $U(X_{DV}) = A$ for some constant A we arrive at the continuous time analogue of the optimal SPRT where the decision variable X_{DV}

can be interpreted as a log-likelihood ratio [18]. Alternatively, one might consider the Ornstein-Uhlenbeck process given by setting $U(X_{DV})$ to $A + BX_{DV}$, which allows the modeler to include primacy or decay effects (earlier information is weighted more, or forgotten) by altering B. U therefore describes the influence of prior information on the inclusion of later information in the decision making process. The diffusion term D is constant presumably as noise is assumed to arise from the stimulus and thus be not be governed by the latent decision variable.

Coupled with this are two thresholds $X_{thr,\pm}$ so that a decision is made when X reaches either $X_{thr,+}$ from below or $-X_{thr,-}$ from above. If each alternative is to be modeled as equally probable, then $X_{thr,-} = X_{thr,+}$. One can introduce variability such as to capture bias in a task by adjusting the thresholds or the initial conditions [18].

One of the major limitations of the DD model in (2.34) is that it only allows decisions between two choices. Attempts to generalise this to multiple choices have been made. These include extending the Multisequential Probability Ratio Test (MSPRT) [18], which is asymptotically optimal, or race models where the decision threshold becomes a curved boundary [89], or in the case of continuous decision making considering a 2-dimensional process within a circle where the point of crossing the circle corresponds to the decision made [90]. On the other hand, both the leaky-accumulator models and the recurrent neural circuit models naturally accommodate multiple choices [17].

The simple linear model predicts longer RTs on error trials, as well as longer-tailed distribution of RTs, than what is observed on perceptual decision making tasks [17]. While this does not capture the data on the RDM task, it does capture a range of data on other decision making tasks with human subjects [17]. Hence being able to relate DD decision making models to neural models is an important step in studying the relationship between neural activity and decision making. Indeed several neural decision making models can be shown to be equivalent to the linear DD model when the parameters are appropriately chosen [18].

Conversely, correlates of several components of the DD model can be found in neurobiology, including the threshold [91]. With neurobiology in mind, R-STDP (as in equation (2.31)) can be shown to allow such a model to achieve the weights necessary to perform MSPRT [92]. Thus we see how synaptic plasticity, neurobiological modeling, and the theory of decision making are all three sides of the same coin.

Furthermore, a pooled-inhibition model, which is similar to a recurrent neural circuit model but which does not necessarily require the slower reverbratory dynamics - and thus with strong synapses and fast activity decay is formally equivalent to an Ornstein-Uhlenbeck model [18] - has been combined with *punishement*-driven plasticity in the Drosophila [93] to explain orienting behaviour.

2.3.3 The Wang Model

The Wang model is a recurrent neural circuit model which stems from [19] where it was originally used to account for how recurrent excitation arising from slow NMDA processes can lead to sustained increased activity. It was adapted in [20] to describe LIP data of monkeys performing the RDM, where it was modeled as receiving an external noisy stimulus as a proxy for area MT inputs to the LIP area, and was found to be a good fit to data. This model thus provides a biologically feasible candidate model for decision making tasks. While it was fitted to the LIP area, it can be used to describe decision making more generally and agnostic of location in the brain [17]. As discussed above, it both captures a wide array of data and can handle more than two choices. However, as seen above it is also not the only biologically feasible option and in fact may not yield optimal performance on some tasks.

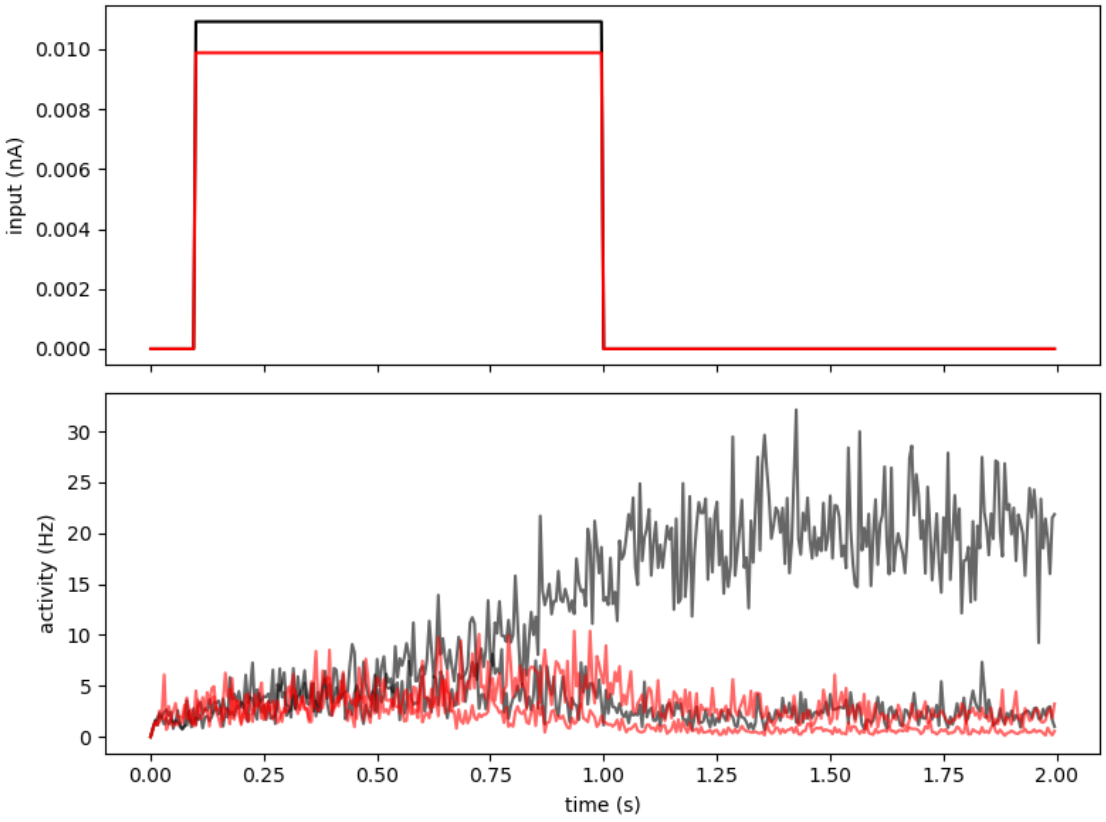


Figure 2.8: Typical trajectories of firing rates in a recurrent neural circuit model. Here the 2-variable reduced model from [94] has two selective populations with two firing rates. A step current corresponding to the mean input of a coherence of 0.1 is provided. Two trajectories of firing rates are shown. The black curves show the firing rate of the population selective for the direction of the coherent dots' movement, while the red curves show the firing rate of the population selective for the opposite direction. During stimulation, slow ramping of activity occurs due to slow NMDAR dynamics followed by competitive inhibition. Persistent elevated activity is not guaranteed. Parameters and code adapted from github.com/xjwanglab/book/blob/master/wong2006/wong2006.py.

The Wang model has a few distinguishing features. In [95] it has been extended to a multiple-circuit model to account for different brain regions involved in decision making, including a bio-

logically feasible implementation of a decision threshold. As can be seen in the Appendix A it can be reduced to a rate-based model, but in [94] it has been reduced to a non-linear 2-dimensional model (shown in Figures 2.8 and 2.9). Here it was studied and shown to exhibit a subcritical bifurcation parameterised by the recurrent synaptic strength w_+ and strength of the stimulus. For a range of w_+ and stimulus strengths, elevated firing rates - or, equivalently, elevated fractions of open NMDAR channels $\langle s_{NMDA} \rangle$ - could be achieved and through hysteresis would persist once the stimulus was removed. A second bifurcation ensured that this effect would not happen if too strong a stimulus was provided. In [88], using weakly nonlinear analysis and focusing on this subcritical bifurcation, this model was further reduced to a 1-dimensional DD model. This 1-dimensional model captures the psychometric features which were explained by the Wang model, and as such successfully finishes bridging the gap between the biologically inspired Wang model and the psychometric decision making studies; however this 1-dimensional nonlinear model does not capture the hysteresis and sustained activity of the prior models, likely due to limiting the analysis to terms no more than cubic in the 1-dimensional decision variable.

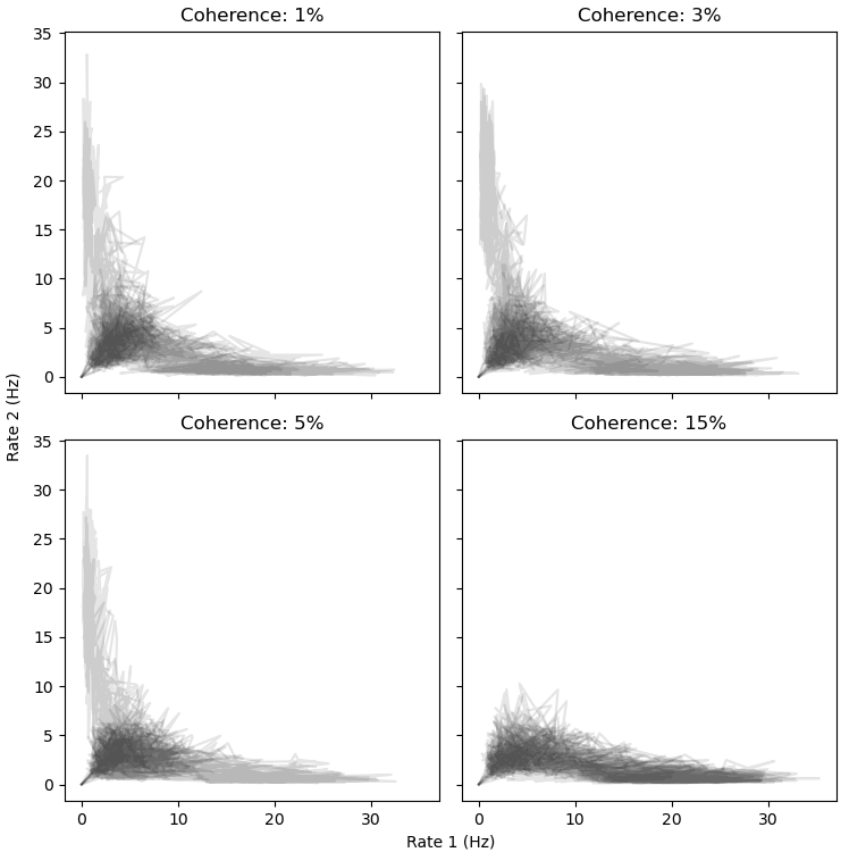


Figure 2.9: Decision making with recurrent neural circuit models can be understood by their dynamics in their state space. Here, the firing rates of two populations develop over time. The x-axis shows firing rates for the populations selective for the direction of coherently moving dots, the y-axis shows firing rates for the population selection for movement in the opposite direction. As the coherence rises and the task becomes easier, the probability of reaching the steady state corresponding to the correct direction selection increases. Multiple trajectories are overlaid. Parameters and code adapted from github.com/xjwanglab/book/blob/master/wong2006/wong2006.py.

One caveat about these reductions is that once we move past the rate-based version considered herein (and derived in the Appendix A), they obfuscate the effect of synaptic weights through function fitting and thus make questions about synaptic plasticity unclear.

The Wang model has also been combined with a reward-driven synaptic plasticity rule [24], which resulted in matching law behaviour consistent with the covariance-driven learning of Loewenstein [83, 84] (as discussed in Section 2.2.5), but this plasticity rule was stochastic and used discrete synapses, and thus is not an example of the plasticity rules considered in this thesis.

As a side note, a two-population rate-based attractor model, or rate-based recurrent neural circuit model, has been combined with the covariance-driven R-STDP rule in [96] to account for behaviour in free-operant experiments where the subject can continuously engage with the environment in a foraging task (rather than trial-to-trial experiments as are typically investigated), and was also able to capture features of observed behaviour such as matching and the exponential distribution of stay durations i.e. the amount of time the subject stays at one site before moving to another.

The components of the Wang model were discussed in Section 2.1. For brevity and reference, the complete model description is included here:

$$\tau_m \frac{dV_i(t)}{dt} = -(V_i(t) - V_L) - \frac{I_i(t)}{g_m} \tag{2.35}$$

$$V_i(t) \leftarrow V_{reset} \quad \text{if} \quad V(t^-) = V_{thr} \tag{2.36}$$

$$\begin{aligned} I_i(t) = &\sum_{j \in \mathcal{P}_E} w_{ij}(V_i(t) - V_E)\ (s_{j,AMPA}(t) g_{AMPA} + s_{j,NMDA}(t) g_{NMDA}(V_i)) \\ &+ \sum_{j \in \mathcal{P}_I} w_{ij}(V_i(t) - V_I)\, s_{j,GABA}(t) g_{GABA} \\ &+ \sum_{j \in \mathcal{P}_{ext}} (V_i(t) - V_E)\, s_{j,AMPA}(t) g_{AMPA,ext} \end{aligned} \tag{2.37}$$

$$g_{NMDA}(V_i(t)) = \frac{g_{NMDA}}{1 + \gamma_{JS} \exp(-\beta_{JS} V_i(t))} \tag{2.38}$$

$$\frac{ds_{j,AMPA/GABA}(t)}{dt} = -\frac{s_{j,AMPA/GABA}(t)}{\tau_{AMPA/GABA}} + S_j(t) \tag{2.39}$$

$$\frac{dx_{j,NMDA}(t)}{dt} = -\frac{x_{j,NMDA}(t)}{\tau_{NMDA,rise}} + S_j(t) \tag{2.40}$$

$$\frac{ds_{j,NMDA}(t)}{dt} = -\frac{s_{j,NMDA}(t)}{\tau_{NMDA,decay}} + \alpha x_{j,NMDA}(t)(1 - s_{j,NMDA}(t)) \tag{2.41}$$

The first equation (2.35) along with the reset condition (2.36) give the LIF membrane potential dynamics for the focal neuron i, while (2.37) gives the synaptic input current to neuron i. The Jahr-Stevens formula for NMDAR dependence on membrane potential is in (2.38). Equations (2.39), (2.40) and (2.41) describe the fast (AMPAR and GABA_A Receptor) and slow (NMDAR) dynamics

of the ion channels. $\mathcal{P}_E$, $\mathcal{P}_I$ and $\mathcal{P}_{ext}$ collect the indices of excitatory (pyramidal), inhibitory (interneuron) and external Poisson neurons synapsing onto the focal neuron i, respectively. Coupled with a plasticity rule, the full model would have the synaptic strengths w_{ij} be plastic as well.

2.4 Evolutionary Algorithms

EAs are a subset of the study of biologically inspired algorithms, explored primarily today in the realms of artificial intelligence and operations research. They provide a broad, general purpose solution to the task of global optimisation. Given a fitness function Φ to be optimised - such as maximising reward on an RL task, or minimising travel time in a traveling salesman problem - an EA strictly needs nothing more than a means by which to order candidate solutions, or genotypes, γ within a population Γ by their fitness scores $\Phi(\gamma)$ to be able to determine the globally optimal candidate [97, 13]. By comparison, RL usually require evaluative feedback $\Phi(\gamma)$ while supervised learning methods require instructive feedback $\gamma^* = \operatorname{argopt} \Phi(\gamma)$ [82]. Moreover, EAs do not require gradient information of the fitness function. That said, the more information provided, the better the EA can be made to perform. Thus, the task of finding an optimal plasticity rule for an uncertain RL task is a prime candidate problem for an EA.

In what follows, I will give the necessary background of EAs and in particular that of CMA-ES, sufficient to understand the choices that need to be made in evolving a plasticity rule. Next I shall discuss some other approaches, such as neuroevolution.

2.4.1 Background

Initially, the study of EAs was composed of three distinct branches: the study of Genetic Algorithms (GAs) , the study of Evolutionary Programming (EP) and the study of Evolution Strategies (ESs) [97, 13]. Later these algorithms were combined under a general framework [13] consisting of the repeated application of various evolutionary operators: the recombination and mutation operators are known together as the variation operators, but there is also the selection operator which utilises the fitness information, potentially alongside a marriage operator. Below I will omit discussion of marriage operators aside from saying that they allow one to implement speciation effects by creating subpopulations which cannot - or are unlikely to - interbreed.

EAs work by the repeated application of these operators to the population. At generation g the survivors, or parents, for the next generation are determined by applying the selection operator to the population Γ_g. In turn, if necessary, these parents are matched up by the marriage operator. New candidate solutions are produced by applying the recombination operator to the parents, and finally the next generation Γ_{g+1} is obtained by applying the mutation operator to these candidate solutions.

The choice of operators, as well as the choice of representation of the genotypes γ, depends on the task at hand. Such considerations include whether the search space is discrete, such as solving an integer programming task, or continuous, such as finding the minimum of a function on a vector space over the real numbers, or even combinatorial, such as the collection of all permutations of nodes in a graph, and it may depend on whether the search space is bounded or unbounded.

The Encoding

Occasionally a distinction is made between whether the candidate solutions are genotypes or phenotypes. The distinction is rather artificial, but pertains to whether the representation of the candidate solution is in some way transformed (for genotypes) or left as is (for phenotypes) before the selection operation is performed. Collectively, I will call the representation of a candidate solution a genotype and denote it γ.

GAs originally considered the genotype to be a string of digits, from which it was proven that a binary representation is optimal, at least for producing the first new generation [14, 13]. However, it is desirable that nearby[7] genotypes have similar fitnesses so that the EA can make many incremental improvements rather than relying on few chance improvements which might achieve performance little better than a random search. Finding such a binary representation over an interval of real numbers can be challenging: for example, representing numbers in the interval [0,1] by their binary expansions truncated at some n bits shows that

$$\frac{1}{2} = 0.10\ldots0$$

is very far from the nearest representable preceding number

$$0.01\ldots1$$

if the mutation operator implements component-wise changes. It is often advisable therefore when considering a real domain to represent the genotypes with real numbers. When dealing with higher dimensional candidates, genotypes might be represented by vectors.

Another consideration is that of boundaries. If there is a feasible region within which candidate solutions must lie, but which the evolutionary algorithm can escape, one needs to find a way to ensure that only feasible solutions are found. If the space is bounded in a known way, one might pass the candidate solution through a bounded function. For example, when searching for a solution within the interval (0,1), if mutation is determined by adding Gaussian noise then on chance a genotype γ may escape the interval under mutation and yield an infeasible solution; if, for example, the fitness function Φ depends on a logarithm, then $\Phi(\gamma)$ might not be well defined. If we denote

[7] Nearby in the sense that the expected number of mutations required to transform one genotypes into the other is small.

the logistic function by $\sigma_{logistic}$, we are guaranteed that $\Phi(\sigma_{logistic}(\gamma))$ is a well defined solution and the evolutionary algorithm can explore all of the preimage $\sigma^{-1}_{logistic}((0,1)) = \mathbb{R}$ for a solution. Alternatively, penalties might be considered if $\Phi(\gamma)$ is still well-defined. Here one adds a penalty to $\Phi(\gamma)$ if γ is not a viable solution so that this fitness becomes worse than that of the nearest viable candidate, but not so poor that the EA is discouraged from finding the nearest viable candidate.[8]

The Selection Operator

Contrasted with the variation operators discussed below, the selection operator is the manner by which fitness information is fed back into the EA. By design, the mutation and recombination operators should be unbiased to avoid genetic drift that might run against the gradient of the fitness function if the fitness function is differentiable, or more generally might run against the optimal direction for improving performance.

The selection operator takes a population Γ_g and returns the survivors, or parents, for the next generation. In ES notation, this number of survivors is typically denoted μ; I shall use μ_{EA} to avoid confusion with the weight-dependence of the plasticity rules. There are several ways in which the selection operator can be implemented [13]: deterministically, as is typically done with ES where the top μ_{EA} candidates are selected based on their fitness scores, or randomly as is done with the other regimes. The random selection process can be implemented in various ways: for example, tournament selection chooses a random subset of candidates and of them selects the best[9] and repeats this procedure μ_{EA} times, while roulette wheel selection assigns each candidate a subinterval of the unit interval [0,1] in proportion to their share of the cumulative population fitness $\sum_{\gamma\in\Gamma}\Phi(\gamma)$ (such that all the intervals are disjoint) and then samples μ_{EA} random numbers from the unit interval and selects the candidates with the intervals within which the random numbers fell. Measures can be taken to avoid repeated sampling. There are certain equalities between these methods, as well as design choices such as the number of candidates chosen for the tournaments or whether the interval sizes should grow nonlinearly in roulette wheel selection with the relative share of the fitness, such as using the squares of the fitnesses $\Phi^2(\gamma)$ as a proportion of the cumulative sum of squares of fitnesses $\sum_{\gamma\in\Gamma}\Phi^2(\gamma)$ for the sizes of the intervals, or if population-performance-dependent baseline is subtracted, such as the minimum fitness score [13].

The Recombination Operator

The recombination operator takes ρ_{EA} parents and combines them to create one or several new candidate solutions. In ES, only one new candidate is created per recombination, while in GAs

[8]The search space, after all, does not strictly need to be connected and passing regions of infeasible solutions might be a prerequisite. Penalties allow the EA to explore the infeasible regions while ideally not stagnating there.

[9]Notice that one only needs to determine which candidate had the best score, not what that score was.

which usually set $\rho_{EA} = 2$, two candidates would then be created. Usually mutation will then be applied to the new solutions, but one might also apply mutation to the parents before recombination, as with Differential Evolution (DE) [98]. Typically the recombination operator takes one of two forms: dominant, or intermediate [99].

In dominant recombination, for each component or gene of the genotypes the value of one of the parents is randomly selected for each child that is being produced. If there are n values - or alleles - for this gene amongst the $\rho_{EA} \geq n$ parents, then the dominant allele is most probably selected. In this way recombination preserves the shared components (which, due to the parents being selected based on fitness, are expected to correlate with the fitness) while randomising the remaining components. In the case of $\rho_{EA} = 2$ when 2 new candidates are created, dominant recombination can be implemented by crossover where the alleles are the parents are shared amongst children.

Intermediate combination is an alternative used when the alleles are continuously valued, as is typical for ESs: a (potentially weighted) average of the parents is taken as a new candidate, to which mutation is then applied. The difference between the average of the parents, which are amongst the μ_{EA} best candidates, and of the population itself is expected to correlate with the gradient of the fitness function, while the spread of the parents orthogonal to the gradient is expected to be averaged away. That is, on average

$$[\nabla\Phi(\langle\gamma\rangle_\Gamma)]^\top \left[\Phi(\langle\gamma\rangle_{\Gamma:\mu_{EA}}) - \Phi(\langle\gamma\rangle_\Gamma)\right] > 0$$

where $\Gamma : \mu_{EA}$ are the μ_{EA} best candidates in Γ. In this way recombination can drive steps in the direction of the gradient of the fitness function (assuming here that Φ is to be maximised; if Φ is to be minimised, the argument is reversed).

In ES literature, an algorithm would be described as a $(\mu_{EA}/\rho_{EA}, \lambda_{EA}) - ES$ or a $(\mu_{EA}/\rho_{EA} + \lambda_{EA}) - ES$, where in the special case of $\rho_{EA} = 1$ the ρ_{EA} is usually omitted. A $(\mu_{EA}/\rho_{EA}, \lambda_{EA}) - ES$ is one where, at each generation $\lambda_{EA} > \mu_{EA}$ new candidates are produced, of which the new μ_{EA} parents are selected. A $(\mu_{EA}/\rho_{EA} + \lambda_{EA}) - ES$ produces $\lambda_{EA} > 1$ new candidates and the best μ_{EA} of the total $\mu_{EA} + \lambda_{EA}$ candidates survive. This latter "+" form implements what is known as *elitism*: the best candidates so far remain in the population. While elitism may have certain advantages by reducing the chance of the population average decreasing, it also makes it more difficult for the population to move away from an elite candidate (especially if they are in a local optimum); as such, although the relative performances of the two types depends on the task at hand, $(\mu_{EA}/\rho_{EA}, \lambda_{EA})$ is recommended for unbounded search spaces [99]. In order to maintain an estimate of the best candidates reached, it is common instead to keep a collection of the best individuals observed throughout the span of the EA, known as a hall of fame.

The Mutation Operator

It is the mutation operator more than anything else that distinguishes the types of EAs [97]. In GAs, most of the variation is designed to come from the recombination operator, while for EP and ESs the mutation operator contributes most of the variability. ESs are further distinguished by evolving simultaneously the genotypes and the parameters for the mutation operator. For brevity I will only discuss mutations over real-valued vectors $\gamma \in \mathbb{R}^n$.

In determining the mutation operator, one might start by considering a few properties that it should satisfy, namely (following [99]):

1. reachability: that any genotype can be reached in a finite number of mutation steps from any other genotype. This is a requisite for proving global convergence.

2. unbiasedness: that the mutation should introduce randomness - or information - into the system as maximally as possible but without bias. Naturally this leads to sampling changes to the alleles from a maximum entropy distribution such as $\mathcal{N}(0, \sigma_{\gamma,l})$ where $\sigma_{\gamma,l}$ describes the size of the mutation applied to the l-th allele. We can denote all these parameters together by a vector σ_γ.

3. scalability: that the mutation operator *itself* can adapt to the landscape.

If the recombination of the parents is written as γ_{recom}, and the offspring after mutation as γ_{offsp}, then we have that

$$\mathbb{E}[\gamma_{offsp}|\gamma_{recom}] = \gamma_{recom} = \sum_{\gamma \in \Gamma : \mu_{EA}} w(\Phi(\gamma))\, \gamma$$

where $w(\Phi(\gamma))$ is a potential reweighting of the average dependent on the fitness of the individual parents.

The scalability property means that the parameter vectors σ_γ should themselves be able to adapt to the landscape, also in an unbiased fashion. For better or for worse, the $\sigma_{\gamma,l}$'s are usually lognormally distributed with mutations coming multiplicatively from $\mathcal{N}(0, \tau_{EA})$ where τ_{EA} is a hyperparameter. This means that we recover this martingale property only for the logarithms of the mutation parameters:

$$\mathbb{E}[\log \sigma_{\gamma_{offsp}} | \sigma_{\gamma_{recom}}] = \log \sigma_{\gamma_{recom}}$$

The adaptation of the mutation parameters allows one to adjust the level-set ellipsoids for sampling vectors of mutations in an axis-parallel manner to locally fit the landscape. Going a step further, CMA-ES computes a population-wide covariance-matrix estimate Σ_g to adapt rotations to the fitness landscape as well.

What is the benefit of this self-adaptation of mutation parameters? This can illustratively

be seen with a simple example: consider the spherical fitness function $\Phi(\gamma) = ||\gamma - r||^2$ to be minimised. The optimum is $\gamma = r$. As the size or euclidean norm of the mutation vectors decreases, the probability that the direction of the mutation vector is towards r increases towards 0.5, but the rate at which the population moves towards r decreases towards 0. Conversely, as the size of the mutation vectors increases, the rate at which the population moves towards r increases but only insofar as the mutation is in the correct direction, which decreases in probability to 0. Thus, at the two extremes, the rate at which the population moves towards the optimum drops to zero. Furthermore, as the population moves closer to r, the mutation sizes need to decrease so as not to overshoot the target. Somewhere in the middle of the extremes is a landscape specific optimum for σ_γ for each γ, or Σ_g for the population, determined by the current location of the population in the landscape.

Benefits of Evolutionary Algorithms

The observation that the globally best candidate will be found for a discrete representation of the genotype γ, or that one will come ϵ-close to a global optimum for continuous representations of γ, depends on the mutation strength remaining sufficiently high that no part of the search space becomes unreachable [12, 97, 13]. Indeed, if the fitness function being optimised for is deterministic i.e. $\Phi(\gamma)$ is a scalar, then sufficient criteria to guarantee convergence to a global optimum are [12]:

1. The probability that there is no optimal candidate in the population Γ_{k+1} conditioned on there being an optimal candidate in population Γ_k is 0. This is satisfied by all elitism strategies, or $(\mu_{EA}/\rho_{EA} + \lambda_{EA}) - ES$ algorithms.

2. If there is no optimal candidate in a population Γ_k, then, loosely speaking, there must be a sufficient probability that Γ_{k+n} contains an optimal solution for some $n \in \mathbb{N}$. Sufficiently strong mutation strength satisfies this condition.

In brief, the set of populations containing optimal solutions must be an accessible and attracting set.

However the global optimisation property offers minimal comfort if one is not able to determine whether a solution ϵ-close to the optimum has been achieved [13]. Practically, the primary benefits of EAs lie elsewhere: firstly, they are massively parallelisable and thus able to utilise multicore processors and high performance clusters efficiently; secondly, as discussed above, they can find solutions to optimisation problems which may be analytically intractable and may not even have differentiable fitness functions.

2.4.2 The Covariance Matrix Adaptation Evolution Strategy

The CMA-ES algorithm is described as the de facto standard in continuous domain evolutionary optimisation [21]. It is an ES of the $(\mu_{EA}/\mu_{EA}, \lambda_{EA})$ type whereby a population-wide covariance matrix estimate is maintained and updated at each generation. The details of this operation lend little to the discussion here, but can be found in Algorithm 5 in [21], as well as in [100] with a thorough discussion.

In broad terms, at each generation g a new population of $\lambda_{EA} > \mu_{EA}$ genotypes are sampled from $\mathcal{N}(\langle\gamma\rangle_{\Gamma_{g-1:\mu_{EA}}}, \Sigma_{g-1})$. The covariance matrix Σ_{g-1} is then updated using the fitnesses of the new candidates as well as the low-pass filtered history of all prior updates (see Figure 2.10). This latter step is justified as follows [100]: on average, mutations without improvement should cancel each other out, yielding short cumulative paths. However, if the cumulative path is long then there is a trend in the successful mutations suggesting a gradient to be followed. Thus the average path length over prior mutations can be compared with the expected path length to determine if it was shorter or longer than average. Conceptually in this way the covariance matrix can adjust to fit valleys in the fitness landscape (see Figure 2.10).

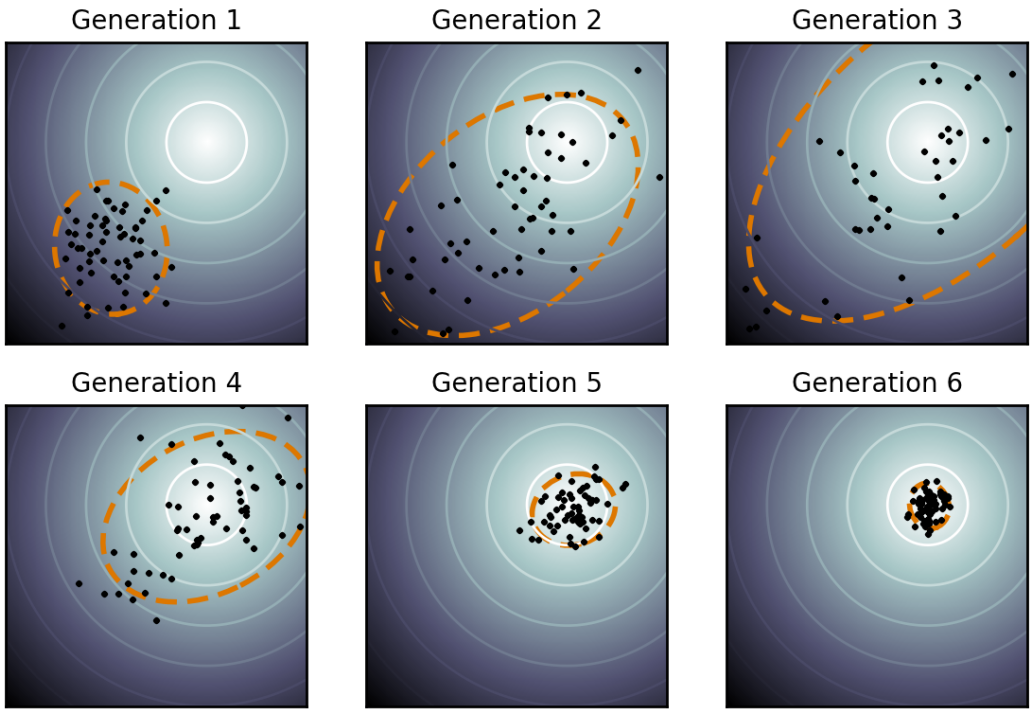


Figure 2.10: CMA-ES progress across several generations. The covariance of the mutation distribution Σ_g adapts in the direction of the gradient of the fitness function. Genotypes from each generation are shown as black dots, while the covariance of the mutation distribution is shown in orange. The fitness function is the spherical fitness function, whose contours are shown in white. Image obtained from en.wikipedia.org/wiki/CMA-ES.

There has been a recent resurgence of interest in CMA-ES following its application in combination with proximal policy optimisation in [101] to successfully solve RL tasks. In short, the

idea from CMA-ES of updating the covariance matrix and then updating the mean was applied to update the policy covariance and then policy mean at each generation. Prior to this, in 2003 in [102] Igel found that on the simple pole-balancing task, using CMA-ES to fit the weights of a neural network with a standard architecture outperformed most other contemporaneous neuroevolution methods. Later in [103] a slightly different algorithm, Natural Evolution Strategies (NES) which adapts the distribution parameters using the natural gradient [104], was found to perform well on a suite of RL tasks when used to fit the weights of neural networks.

2.4.3 Neuroevolution

One of the limitations of using CMA-ES in particular, or ES in general, to determine the parameters of a neural network (be it artificial or biologically inspired) is that these methods do not determine the topology of the network. The discipline of using EAs to design neural networks is known as neuroevolution and it is fraught with difficulties [14, 105] (despite the successes mentioned in the previous section). The first challenge arises from trying to perform crossover with networks of different topologies: one needs to find a representation of the networks such that crossover operations can be performed without bias. The second challenge arises from the complexity of the search space: once the evolved candidates begin to perform well, it becomes remarkably easy for mutations to worsen performance. Unlike the spherical fitness function $\Phi(\gamma) = ||\gamma - r||^2$ mentioned in 2.4.1 where the probability of an improvement can increase to 0.5, it would seem that most potential mutations to the topology of the network will typically worsen its performance on complex tasks.

Remarkable success in evolving network topologies came with NeuroEvolution of Augmenting Topoligies (NEAT) [106], leading to several successful avenues of research [107, 108, 109, 31] and culminating in the idea of an Evolved Plastic Artificial Neural Network (EPANN) [110]. These techniques are strongly oriented towards solving tasks rather than investigating biological models. Nonetheless, due to the success of NEAT in evolving topologies it deserves mention.

In an attempt to determine whether NEAT introduced genetic drift - or rather, to determine if it would be a feasible candidate mechanism to evolve the topology of the Wang model - I ran a small test: I used it with a constant fitness function. The evolved networks were found to have Poisson-like distributions of degrees for the nodes, evident of being of the Erdős–Rényi type [111, 112] but not consistent with biology [113, 114, 60, 115]. Rather than attempting to offset this genetic drift, I opted to maintain the simpler fully connected topology of the Wang model.

2.5 Optimality

Here I wish to address whether - and how - one should use a normative or optimisation-oriented approach to interrogate biology. Such an approach has, after all, been criticised for being too simplified [10]. When biological data does not match optimal performance with respect to some fitness function, typically the researcher would adapt their fitness function to better capture the data. However, this runs the risk of overfitting: how does one determine whether the fitness function is incorrect or if failure to match the fitness function is due to stochasticity in the evolutionary process? One might be able to do this by comparing the observed distribution of data to that predicted by a normative model, but to do so one needs to obtain a distribution from the normative model in the first place.

I will start with a rather spurious argument that lays the conceptual groundwork for more sound examples below.

Imagine a k-dimensional continuous-valued collection of biological traits, represented by χ.[10] The individual components of χ might correspond to heights, fur colour, even something more abstract such as parameters of the reaction-diffusion dynamics that might give rise to fur patterning, or even the layout of cells in an animal body. Now imagine that these parameters adapt to minimise some fitness function Φ,[11] but undergo regular noise in this process which can be parameterised with a strength D, and that the changes to χ that arise over time are sufficiently small that we can model this as diffusive noise. This noise can be independent of the fitness function, so that the noise arises from biological processes while the fitness function is determined by a more abstract external environment. Finally, imagine that the environment changes in such a way that Φ changes on a wholly slower timescale to that of χ such that we can consider a separation of timescales. Then, if the fitness function is integrable $\Phi \in L_1(\mathbb{R}^k)$ and we assume that the drift χ has a potential proportional to Φ we arrive at the dynamics

$$\frac{d\chi}{dt} = -a\nabla\Phi(\chi) + \sqrt{2D}\eta(t)$$

where η is a white noise process and a is a proportionality constant, or

$$d\chi = -a\nabla\Phi(\chi)dt + \sqrt{2D}dW_t$$

where W is a k-dimensional Wiener process.

[10] I am avoiding the use of γ to not confuse the actual biological traits with any notion of a genotype. χ can be understood as short for "characteristic".

[11] The same argument can be made for maximising a fitness function, save that the fitness function becomes the negative potential of the drift of the traits.

The corresponding Fokker-Planck equation is then

$$\frac{\partial p}{\partial t} = a\nabla \cdot (\nabla \Phi p) + D\Delta p$$

where Δ is the Laplace operator, and since $\Phi \in L_1(\mathbb{R}^k)$ it follows from Fokker-Planck theory [116] that the equilibrium distribution p_{SS} can be characterised by the Boltzmann distribution

$$p_{SS}(\chi) = \frac{1}{Z} \exp(-a\Phi(\chi)/D)$$

where $Z = \int_{\mathbb{R}^k} \exp(-\Phi(\chi)/D) d\chi$ is the normalising constant. From this we also get

$$\nabla \log p_{SS}(\chi) \propto -\nabla \Phi(\chi) \tag{2.42}$$

This distribution, p_{SS}, gives us a normative estimate of the true biological distribution, and importantly it has modes at the local minima of Φ. There are two ways in which we can use this. First, if we estimate $\chi_{est} = \text{argmin}_\chi \Phi(\chi)$ we can assume that a non-negligible portion of the probability mass lies in a region around χ_{est} and that the logarithm of the probability decreases at a rate proportional to the negative of the fitness function's gradient in that direction. Thus from estimating the fitness function we can perform inference about the biology, an idea taken from [10] which I call the *optimality principle* and will discuss below. Secondly, we have the idea of the *optimality prior* [11]: imagine we are fitting a biological model M dependent on parameters θ_M to describe some dynamic continuous biological phenomenon which addresses a fitness function Φ as before. Now the same reasoning can be applied to the model's parameters to find a prior distribution of the "true" parameters. This prior can in turn be combined with the likelihood of the data to determine a posterior distribution or used as a regulariser to determine a maximum a posteriori estimate of parameters.

Brought together, the value of using normative approaches to fit biological models may lie in determining an estimate of the distribution of parameters of the model and an estimate of the distribution of fitnesses.

2.5.1 The Optimality Principle

In [10] it is shown that if Φ is the wiring cost and χ is the placement of neurons in the *Caenorhabditis elegans* (*C. elegans*) then the optimality principle yields a good estimate of the wiring cost distribution. In *Escherichia coli* (*E. coli*), if Φ is biomass production the principle accounts for deviations of metabolic fluxes from that which maximises biomass production. Moreover, they find that deviations from optimality are larger in dimensions that have less impact on the fitness, as suggested by equation (2.42). Finally, they provide a Bayesian approach to better estimating the fitness function from the data. This in principle allows the researcher to determine which

optimisation criterion is providing the evolutionary drive.

It should be noted that they arrive at the distribution differently to the Fokker-Planck approach above, by finding that the probability mass function over discrete states follows

$$\mathcal{P}(\chi) = f(\Phi(\chi)) \tag{2.43}$$

for some increasing function f. They consider a noisy fitness function $\Phi_{noisy}(\chi) = \Phi(\chi) + \text{noise}$ and consider that the probability that the system is in state $\mathcal{P}(\chi_i)$ is equal to the probability that the noisy fitness for χ_i was greater than for all other states, i.e.

$$\mathcal{P}(\chi_i) = \mathcal{P}(\Phi_{noisy}(\chi_i) > \Phi_{noisy}(\chi_j) \, \forall j \neq i)$$

From the assumption that the noise is independent of the fitness, it follows that there is some increasing f such that Equation (2.43) holds. Importantly, this equation does not require continuous variables. However, one still needs to characterise the distribution of the noise for the noisy fitness function Φ_{noisy} to explicitly determine the distribution from the fitness function.

2.5.2 The Optimality Prior

Optimisation priors are maximum-entropy priors determined from the optimisation criterion Φ [11]. When fitting models to data, theorists often extend the maximum-likelihood approach with the use of uninformed regularisers, which yields a maximum a posteriori estimate and can reduce overfitting. However, if one knows what the modelled system is aiming to achieve, one can use that information to construct a more informative prior with which to regularise the model. Formally, the parameters $\hat{\theta}_M$ for a model M may be chosen with regularisation as

$$\hat{\theta}_M = \text{argmax}_{\theta_M} \log \mathcal{P}(\text{data}|M, \theta_M) + \underbrace{\log \mathcal{P}(\theta_M|M)}_{\text{regulariser}}$$

while using the optimality prior one replaces the uninformed regulariser with one informed by the fitness function:

$$\hat{\theta}_M = \text{argmax}_{\theta_M} \log \mathcal{P}(\text{data}|M, \theta_M) + \log \mathcal{P}(\theta_M|\Phi, M, \beta)$$

where $\mathcal{P}(\theta_M|\Phi, M, \beta)$ is a distribution with parameter β - called the "optimisation parameter" - which influences its entropy. In particular, they consider the Boltzmann distribution

$$\mathcal{P}(\theta_M|\Phi, M, a) = \frac{1}{Z} \exp(-\beta \Phi(\theta_M))$$

which we see, by choosing $\beta = a/D$, is the steady-state distribution p_{SS} of the Fokker-Planck dynamics discussed above. To perform Bayesian inference, one can then sample $\hat{\theta}_M$ from the

resultant distribution rather than choosing a maximiser.

2.6 Conclusion

In this chapter we have covered a range of different topics providing background for the chapters to follow. All of the plasticity rules can be unified into a single family of rules paramterised by continuous variables. In an normative approach, this family of rules can be explored by an EA which attempts to find optimal parameters for the Wang model to learn to solve the RDM task. This is discussed next in the Chapter 3.

Chapter 3

Methods

> One branch of neuroscience that can help bridge findings from the microscopic to the whole-brain level is computational neuroscience.
>
> The Brain Facts Book, [117]

In this section I define rate-based version of the Wang model to be used in simulating the decision making process on the RDM task and to attempt decision making on an XOR task. The XOR task is included to assess the expressivity of the model. For the RDM task, the Wang model is extended with a plasticity rule, parameterised by continuously-valued parameters, which change the synaptic strengths of the model in response to the activity of the populations of neurons and in response to reward obtained from the performance on the task.

I use the EA CMA-ES as an optimisation procedure to find separately parameters for the plasticity rule, and weights for the network in the absence of any synaptic plasticity, so as to allow the network to solve the RDM task. For the XOR task, I only evolve the synaptic weights. Performance on the task at hand thus describes a fitness function, or fitness landscape, over the weights and over the plasticity rule parameters.

Because of the many interactions of the parameters, CMA-ES was chosen due to its ability to adapt to the fitness landscape and capture locally the effect of these interactions of the parameters. Moreover, the optimal parameters for the plasticity rule are likely of very different scales, where some are scaled by the cube of the firing rates and others are constant; CMA-ES is able to adjust the scales of the mutation operation to accommodate this mismatch in scale. The search space is, for most of the parameters, unbounded and as such a $(\mu_{EA}/\rho_{EA}, \lambda_{EA})$ strategy is recommended, and the CMA-ES algorithm falls within this class.

Due to the need of the EA to run many successive simulations, these simulations need to be made as fast as possible. Moreover, reducing the dimensionality of the search space for the EA (in our case, reducing the number of parameters for the plasticity rule while maintaining its generality) can reduce the running time of the EA as well as allow the procedure to find a unique optimal solution rather than a (pseudo)-random point on an optimal surface. To this end a rate-based version of the Wang model will be used, along with the rate-based plasticity rule which one arrives at through averaging. The full derivation of the rate-based Wang model can be found in the Appendix Section A.2.

In what follows, first the complete framework is presented in Section 3.1. The translation of the RDM task to a rate-based framework is adopted from [94] and described in Section 3.2, while the XOR task is described next in Section 3.3. Afterwards, in Section 3.4, I will discuss how the plasticity framework can be extended to include all of the biologically feasible modifications that were discussed in Section 2.2 while at the same time reducing the number of parameters needed to do so by grouping monomial coefficients, followed by how this can include the three-factor formalism of Section 2.2.5 and finally consider the bounds on some of these parameters. The CMA-ES does not inherently accommodate bounds; to circumvent this, genomes are sampled from an unbounded space and bounded at evaluation time. Next I will discuss how the implementation was further accelerated and rendered more stable in Section 3.5 using function approximations and other changes. Finally, implementation details are given in Section 3.6. The final parameters for the Wang model can be found in Table A.1, and the parameters for the plasticity rules and their evolution can be found in Table A.2.

3.1 The Complete Framework

The model used to perform computations is a recurrent neural circuit model with three recurrent excitatory populations (for the RDM task) or five or more recurrent excitatory populations (for the XOR task) and one inhibitory population, as well as external excitatory populations providing sufficient input to maintain a low firing nearly-zero firing rate, as shown schematically in Figure 3.1. In the RDM task, two of the excitatory populations are selective for the two directions of the coherent subset of moving dots, and the saccade direction selection is determined by the firing rates of these populations. In the XOR task, two of the excitatory populations receive the input and another two are used to determine the output. In both cases the final excitatory population and the inhibitory population maintain competitive dynamics between the populations.

From each excitatory population onto every population, including recurrently onto themselves, are collections of synapses with shared synaptic strengths. These represent the average synaptic strength between neurons within the two populations, as are obtained by the averaging discussed below. These average synaptic strengths may undergo plasticity described by a parameterised

plasticity rule which depends on the firing rates of the two populations of the neurons, as well as the current value of the average synaptic strengths and a low-pass filtered history of the firing rate of the neurons on the postsynaptic side.

Two distinct optimisation procedures will be run for the RDM task, both using CMA-ES. The first is to determine the optimal baseline performance of the model on the task for various coherence values. In this procedure, the plasticity rule implements no change and the synaptic strengths are directly evolved. The second optimisation procedure is over the parameters of the plasticity rule, and aims to determine a plasticity rule which, across successive trials of the RDM task, can drive the network to optimal performance.

The full experimental framework is shown in Figure 3.2, including the libraries used in Python for their implementation.

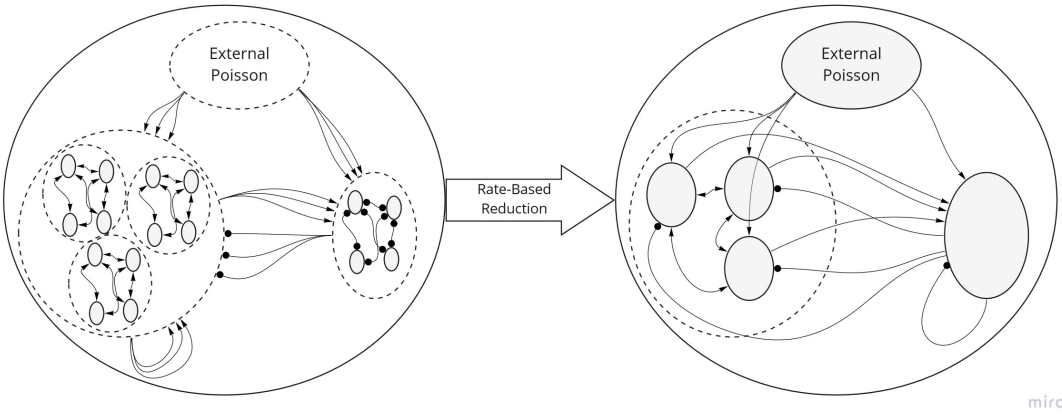


Figure 3.1: The reduction of the spiking model to a rate-based model. Grey nodes show individual dynamic variables: in the spiking model (left), these variables correspond to fractions of open ion channels and membrane potentials; in the rate model (right), these variables correspond to population firing rates and average fraction of open ion channels. In the spiking model, individual Poisson inputs are simulated while in the rate model average Poisson inputs are simulated. Lines with arrowheads indicate excitatory connections, and lines with circles indicate inhibitory connections. Two-headed arrows are used to show bidirectional connections for clarity. Bundles of arrows show diffuse all-to-all connections. Conceptual groupings are shown with dashed lines. In both cases there are three excitatory populations and one inhibitory population, all receiving external Poisson inputs. The populations shown correspond to the model used on the RDM task, while the XOR task used more excitatory populations. Image created using miro.com

3.2 The RDM Task

The individual RDM task trials were modelled after the implementation in [94] for their reduced model. There will be multiple successive trials, each using the weights and time-averaged firing rates θ from the end of the previous trial as the initial conditions for the next simulation. When evolving the plasticity rules, there will be 100 successive trials. When evolving the synaptic weights directly 10 successive trials were used.

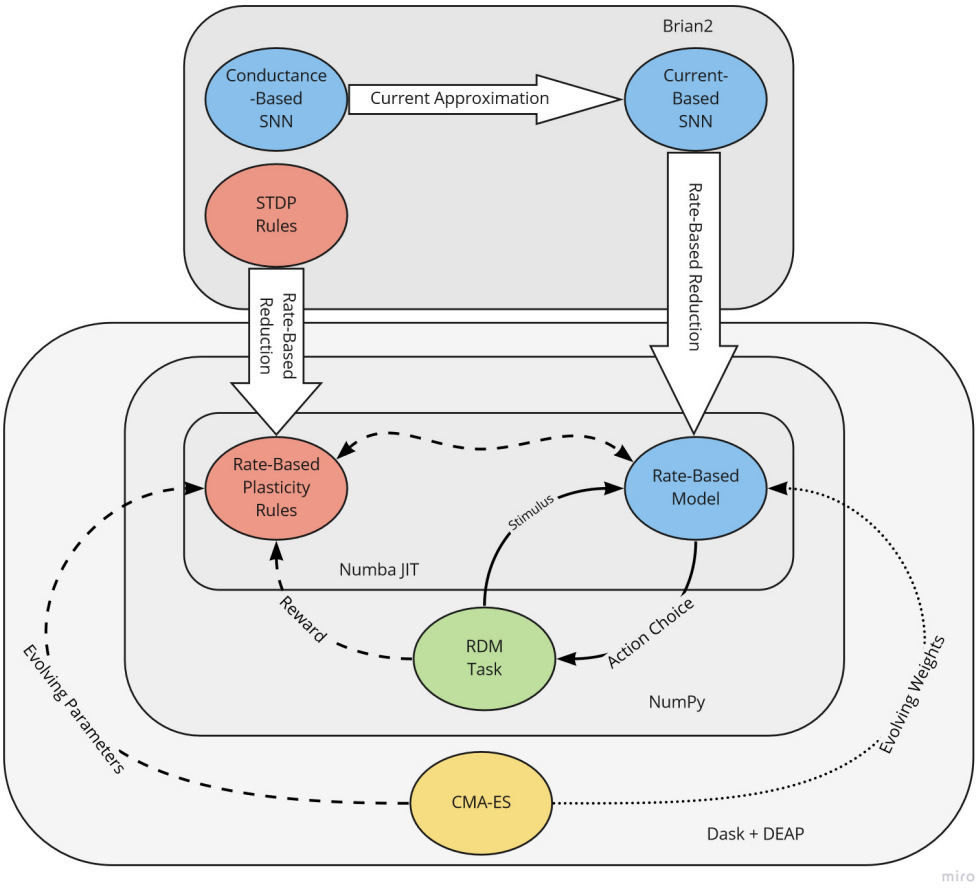


Figure 3.2: The complete framework. Ellipses show distinct theoretical components, while thin arrows between them show interactions between the components and thick arrows show reductions and approximations. The dotted thin arrow shows interactions only present when evolving weights, and the dashed thin arrows show interactions only present when evolving plasticity parameters. Rectilinear boxes show software which was used in simulating each component. All spiking model simulations were implemented using Brian2. The rate-based plasticity rules and Wang model were sped up using Numba's Just-In-Time compilation, which interacted with the RDM task which was implemented in NumPy. This was all called from the CMA-ES algorithm implemented in DEAP and parallelised with Dask. Image created using miro.com

The task took the following form: the simulation was run for 200ms before any stimulus was presented, then the stimulus was presented for a another 200ms, at the end of which the final firing rate was read out as the decision variable. All of these times are in principle variable. The population with the highest firing rate corresponded to the choice made. The stimulus was provided as an increase in firing rate of the external inputs to two of the selective populations corresponding to the two options for the saccade. The increase in the external firing rate was by a fraction $0.05 \times (1 \pm c)$ where c is the coherence of the RDM task, and the $\pm$ reflects whether the population is receptive to activity in the same direction or the reverse. When the decision was made, reward was immediately given, at an amount of $R(0) = 1/\tau_{reward}$ and decayed exponentially with the time-constant $\tau_{reward} = 1$ms i.e.

$$\frac{dR}{dt} = -\frac{R}{\tau_{reward}}$$

Afterwards, the simulation was continued (without stimulus) for another 400ms, which also ensured that the integrated total reward $\int R(t)dt$ reached 1. This meant that the maximum possible score, or sum of integrated rewards, matched the number of successive trials.

To offset noise, the fitness of a genome was determined by the average of several restarts of these successive trials; 10 restarts were used, as initial attempts to use fewer showed little capacity to evolve solutions with non-negative fitness due in part to trial-to-trial variability (cf. Figure 4.2). These restarts would reset the initial conditions to their original values.

As a consequence of the rate-reduction, and in particular the linearisation of the Jahr-Stevens formula around the average membrane potential, division-by-zero errors may occur if the input to a population of neurons is too strong, resulting in NaNs.[1] If a trial resulted in producing a NaN, all reward on that trial was disregarded and the candidate genotype received a penalty of 0.5 for that trial and all remaining successive trials (since usually this resulted in NaN synaptic strengths and θ, rendering the initial conditions of successive trials invalid).

3.3 The XOR Task

In order to test the expressivity of the Wang model, I also considered an implementation of the XOR task. In this case, $p \geq 4$ selective excitatory populations were used, alongside one non-selective excitatory population and one inhibitory population. The first two selective populations were provided with a stimulus as in the RDM task, however the stimulus could be strong or weak for either or both populations corresponding to the four possible XOR inputs: (1,1), (1,0), (0,1) and (0,0). The firing rates of the final two selective populations were interpreted as the estimated XOR output for the stimuli: if the second to last population had a higher firing rate than the final population, then the model predicted XOR of 1; otherwise, a XOR of 0 was predicted.

A range of values between 4 and 8 for the parameter p were tried. The rewards and other features of the trial were identical to that of the RDM task.

3.4 Extending the Plasticity Framework

As mentioned in Section 2.2 and discussed in [22, 23, 63], averaging over spike trains yields rules of the form given in (2.22) and (2.23). Because of the asymmetry in the denominator of the higher order term in (2.23), as well as the inverse dependence on firing rates, the rational functions in (2.23) are much less tractable to including higher-order terms and grouping coefficients (as I will

[1] Removing this limitation is discussed in Section 5.2, but due to time constraints was not achieved here.

do below) than the simpler polynomials in (2.22). We can start by rewriting equation (2.22) as

$$\left\langle \frac{dw_{ij}}{dt} \right\rangle = \theta^p \overline{F}^{X,Y} \nu_j \nu_i + \overline{G}^{X,Y,Y} \nu_j \nu_i^2$$

where the exponent of θ has been replaced with $p \geq 1$ and the products of the Volterra kernel coefficients and decay times into single terms with overlines for clarity: that is, $\overline{F}^{X,Y} = F^{X,Y}\tau_{F,X,Y,0}$ and $\overline{G}^{X,Y,Y} = G^{X,Y,Y}\tau_{G,X,Y,Y,0}\tau_{G,X,Y,Y,1}$. Setting $p \geq 1$ satisfies the requirement of BCM theory of a superlinear dependence on θ.

Now we can expand the plasticity rule to include lower and higher order Volterra kernels (all of the exponential decay type of (2.14), (2.15) and (2.21)) and arrive at

$$\left\langle \frac{dw_{ij}}{dt} \right\rangle = \theta^p \overline{F}^{X} \nu_j + \theta^p \overline{F}^{X,Y} \nu_j \nu_i + \theta^p \overline{F}^{X,X,Y} \nu_j^2 \nu_i + \theta^p \overline{F}^{X,X} \nu_j^2 + \overline{G}^{Y} \nu_i + \overline{G}^{Y,Y} \nu_i^2 + \overline{G}^{X,Y} \nu_j \nu_i + \overline{G}^{X,Y,Y} \nu_j \nu_i^2 + \dots$$

Grouping the monomial coefficients and dropping the $\overline{F}, \overline{G}$ notation in favour of a more concise $\overline{\xi}$ notation gives us

$$\left\langle \frac{dw_{ij}}{dt} \right\rangle = \sum_{ab \in \{01,02,10,20,11,12,21\}} \left(\overline{\xi}_0^{ab} + \overline{\xi}_1^{ab} \theta^p \right) \nu_j^a \nu_i^b \tag{3.1}$$

where I have truncated the expansion at third degree multivariate monomials and second degree univariate monomoials, as higher-order correlations between the pre- and postsynaptic rates seem unlikely to be important in learning to solve the RDM task. Here the superscripts of the variables $\overline{\xi}$ describe the monomial coefficients which they gather i.e. $\overline{\xi}_0^{01} = \overline{G}^{Y}$ and $\overline{\xi}_0^{11} = \overline{G}^{X,Y} + \overline{F}^{X,Y}$ and so forth. The presence of terms such as $\overline{\xi}_1^{12}\theta^p = \left[\overline{G}^{X,Y,Y} + \overline{F}^{X,Y,Y}\right]\theta^p$ admits interactions between the postsynaptic kernel coefficients G and the postsynaptic low-pass filtered firing rate θ which had been absent until now.

These 14 coefficients and the power p, as well as the time constant τ_θ, begin to show the benefit of using an automated optimisation procedure over an iterative manual fitting procedure. As the number of parameters and their interactions grows, a manual fitting procedure would become increasingly difficult and tedious.

To arrive at a fair reduction of the STDP rules discussed in [16] and Section 2.2.3, we also need to include weight-dependence. Since the weight dependence is conditioned on whether the effect is potentiating or depressing, while the EA will allow for both positive and negative ξ values, we need to introduce the functions

$$\xi_k^{ab}(w_{ij}) = \begin{cases} (w_{max} - w_{ij})^\mu \, \overline{\xi}_k^{ab} & \text{if} \quad \overline{\xi}_k^{ab} \geq 0 \\ w_{ij}^\mu \, \overline{\xi}_k^{ab} & \text{otherwise} \end{cases}$$

This includes the extra parameter $\mu \in [0,1]$ over which to optimise. This allows the inclusion of the weight dependence of equation (2.24) while also allowing presynaptic activity to lead to potentiation and postsynaptic activity to lead to depression (as is sometimes observed [16]).

Next we consider weight decay. As discussed in Section 2.2.4, weight decay is a core component of some learning rules, such as Oja's rule. Occasionally STDP rules will add a constant term to capture such weight decay, effectively extending (2.10) to

$$\frac{dw_{ij}}{dt} = X(t)\mathcal{F}(X,Y;\theta,w_{ij}) + Y(t)\mathcal{G}(X,Y;w_{ij}) + C(\theta,w_{ij}) \tag{3.2}$$

where the dependencies on the postsynaptic firing trace θ as well as on w have been included. To allow for multiplicative decay with dependence on the postsynaptic firing rate as discussed in Section 2.2.4, we can choose C of the form

$$C(\theta,w_{ij}) = \xi^{00}\theta^{p_{decay}}w_{ij}$$

The mean rate of change of the synaptic weights is thus

$$\left\langle \frac{dw_{ij}}{dt} \right\rangle = \xi^{00}\theta^{p_{decay}} \langle w_{ij}\rangle + \sum_{ab\in\{01,02,10,20,11,12,21\}} \left(\xi_0^{ab}(\langle w_{ij}\rangle) + \xi_1^{ab}(\langle w_{ij}\rangle)\theta^p\right)\nu_j^a\nu_i^b \tag{3.3}$$

Notice that with $p_{decay} = 2$ and $\theta \xrightarrow{\tau_\theta \to 0} \nu_i$ we obtain the weight decay of Oja's rule, namely $\xi^{00}\nu_i^2 w_{ij}$ with $\xi^{00} < 0$, while in general varying p_{decay} allows for a range of multiplicative (biologically feasible) weight decay rules.

Currently we have 19 free parameters, but I will introduce 2 more.

3.4.1 Including the Three-Factor Formalism

The next step is to consider the three-factor formalism. As we saw in Section 2.2.5, if we include dependence on a reward signal we can solve reinforcement learning tasks and approximate the dependence of plasticity on a neuromodulatory signal which encodes the reward. Starting from (2.31), if we make the assumption that the reward signal R is conditionally independent of the neural activity (conditioned on the action selected) and hence conditionally independent of learning rule H, we can average over the eligibility trace e_{ij} in (2.31) separately from reward and we arrive at the equations

$$\begin{aligned} \tau_e \left\langle \frac{de_{ij}}{dt} \right\rangle &= -\langle e_{ij}\rangle + H_2(\nu_j,\nu_i;\theta,\langle w_{ij}\rangle) \\ \left\langle \frac{dw_{ij}}{dt} \right\rangle &= \langle e_{ij}\rangle\ R \end{aligned} \tag{3.4}$$

where I have omitted "conditioned on action selection" notation and included the rates $\nu_{j/i}$ in H_2 in place of the spike trains.

However, unsupervised learning happens in the absence of reward signal. This may be thought of as the network fitting the data distribution in absence of reward; a highly separable representation of the data often allows for faster learning when the reward or supervisory signal does arrive. To allow for such unsupervised learning, I introduce the parameter $\beta \in [0, 1]$ and use the dynamics

$$\begin{aligned} \tau_e \left\langle \frac{de_{ij}}{dt} \right\rangle &= -\langle e_{ij} \rangle + H_2(\nu_j, \nu_i; \theta, \langle w_{ij} \rangle), \\ \left\langle \frac{dw_{ij}}{dt} \right\rangle &= \langle e_{ij} \rangle \; R\,(1-\beta) + \beta H_2(\nu_j, \nu_i; \theta, \langle w_{ij} \rangle) \end{aligned} \tag{3.5}$$

where H_2 matches the right-hand side of (3.3):

$$H_2(\nu_j, \nu_i; \theta, \langle w_{ij} \rangle) = \xi^{00} \theta^{p_{decay}} \langle w_{ij} \rangle + \sum_{ab \in \{01,02,10,20,11,12,21\}} \left(\xi_0^{ab}(\langle w_{ij} \rangle) + \xi_1^{ab}(\langle w_{ij} \rangle) \theta^p \right) \nu_j^a \nu_i^b$$

For full generality, the two instances of H_2 might depend on different parameters but because this would nearly double our search space for the optimisation procedure, I have opted to use the same parameters for both instances. This adds τ_e and β to the other 19 parameters for the EA.

3.4.2 Constraints on Parameters

Some of the parameters for the learning rule are bounded: the time constants τ_e, τ_θ need to be strictly positive, while to allow for BCM-like effects p should be greater than 1. μ and β should be between 0 and 1. Yet the EA CMA-ES has no internal notion of bounds i.e. it performs an unbounded search in the parameter space. To avoid infeasible parameters, I selected genotypes (or candidate individuals) from within $\mathbb{R}^{21}$ and constrained the values corresponding to these learning parameters by passing the genome through the softmax function (for those bounded below) and sigmoid function (for $\mu, \beta \in [0, 1]$). Thus, the search space was $\mathbb{R}^{21}$ but only feasible learning rule candidates were selected.

When evolving the weights directly, a scaled version of the softmax function was used to keep the resultant weights in the interval $(0, w_{max})$.

For reference, the collection of evolved parameters can be found in the Appendix Table A.2.

3.5 Accelerating Dynamics

The derivation of the mean-field model for a conductance-based spiking neural network, and the consequent firing rate model, found in [36] has a computational limitation: at each timestep of the simulation, one needs to self-consistently compute the firing rate update, the "effective" parameters (such as the effective membrane time constant), and the average membrane potential, all of which depend on one other. In my initial attempts to implement this, I arrived at a simulation several times slower than the spiking model. To speed it up, I drew inspiration from [94]: Firstly, I replaced the conductance-based model with a current-based approximation by replacing the driving forces of the internal connections with fixed values. This yields a mean-field model where the "effective" parameters are once again constant. Secondly, because quadrature integration of the inverse first-passage time formula (A.18) proved to be slow, I fitted an approximate firing rate curve of the same form as discussed in [94] and [118], but with the distinction that I include dependence on the noise strength. This has the marginal benefit that the resulting simulations are more numerically stable. Finally, unlike in [19] where they use only the asymptotic values of the synaptic gating variables, and [36] where it is suggested to treat only the average NMDA-gating variable $\langle s_{NMDA} \rangle$ as dynamic (due to its slower timescale), I treat all synaptic gating variables as dynamic. This benefitted from truncating the formula for the asymptotic fraction of open NMDAR channels ψ.

The resulting model differs from the first model mentioned in [94] by including noise strength dependence in the firing rate curve, by including the Jahr-Stevens formula linearised around the estimated average membrane potential, and finally by using the truncated ψ rather than the value for regular firing.

3.5.1 Current-Based Approximation

To convert the conductance-based model to a current-based model, I replaced $V_i(t)$ the driving forces for the synaptic inputs $V_i(t) - V_{E/I}$ with a constant V_{drive} i.e. the driving force became $V_{drive} - V_E$ for the AMPA-mediated excitatory inputs, $V_{drive} - V_E^{eff}(\langle V_i \rangle)$ for the NMDA-mediated excitatory inputs and $V_{drive} - V_I$ for the GABA-mediated inhibitory inputs. Several arbitrary values were tried for V_{drive}, increasingly from around the true average membrane potential of the non-selective excitatory cells near -55mV, and ultimately a value of $V_{drive} = -47.5$mV was chosen as increasing V_{drive} brought the mean and standard deviation of the resultant membrane potential distribution of the current based model closer to that of the conductance based model (see Figure 3.3). However, as V_{drive} increased too much, the current-based model lost some qualitative features of the original model. Particularly, the steady states of elevated activity seemed to disappear.

3.5.2 Approximate Firing Rate Curve

Following [94, 118], I approximated the firing rate of the inhibitory population with a curve

$$\hat{\phi}(I) := \frac{-c_I I - \overline{I}_I}{1 - \exp\left[-g_I\left(-c_I I - \overline{I}_I\right)\right]} \approx \phi(I), \tag{3.6}$$

where I is the synaptic current flowing out of the cell as given in (2.37), and $c_I, \overline{I}_I, g_I$ are parameters. The discrepancy with the equation given in [94] arises from them using current flowing *into* the cell. Optimal values of $c_I, \overline{I}_I$ and g_I were found using scipy's `fmin` function and minimising the mean-squared-error between the true firing rate curve and the resultant approximation. See Figure 3.4.

Because the external Poisson inputs to the excitatory populations can change (as a consequence of providing an input signal), and consequently so too does the membrane potential noise, I fitted a family of functions for the excitatory firing rates dependent on membrane potential noise σ_V. That is,

$$\hat{\phi}(I, \sigma_V) := \frac{-c(\sigma_V) I - \overline{I}(\sigma_V)}{1 - \exp\left[-g(\sigma_V)\left(-c(\sigma_V) I - \overline{I}(\sigma_V)\right)\right]} \approx \phi(I, \sigma_V), \tag{3.7}$$

where $c, \overline{I}, g$ are given by polynomial expansions

$$\begin{aligned} c(s) &= \sum_{k=0}^{4} a_{c,k}\, s^k \\ \overline{I}(s) &= \sum_{k=0}^{4} a_{I,k}\, s^k \\ g(s) &= \sum_{k=0}^{4} a_{g,k}\, s^k \end{aligned} \tag{3.8}$$

The degree of the polynomials was arbitrarily limited to 4. Polynomials were used for ease of computation, avoiding expensive numerical operations that might be present in more typical basis functions. Fitting the polynomials was done by sampling various synaptic noise strengths σ and using scipy's `fmin` function to compute optimal (with respect to mean-squared-error) values of $c, \overline{I}$, and g for the various values of σ. See Figure 3.6 for the fitted polynomials. The fit is not perfect (as can be seen in Figure 3.5) but this likely arises from the complexity of the curve the polynomials are fitting.

Due to numerical imprecision when computing true firing rates above a point using quadrature integration (notice the kink in the curve in 3.4), as well as due to the fact that approximate curve might not capture the full complexity of the true curve, all curve fitting was done in the range of firing rates from 0.1Hz to 200Hz. Most firing rates reside within these bounds. Finally, as a final precaution, the approximated firing rate curves were set to return $1/\tau_{refrac}$ if the approximate curve would yield a higher rate.

Numerical Stability

The true firing rate formula, or the inverse first-passage time formula, contains an integral where the integrand contains the error-function:

$$\phi(I,\sigma) = \left[\tau_{refrac} + \tau_m\sqrt{\pi}\int_{(V_{reset}-V_{SS})/\sigma}^{(V_{thr}-V_{SS})/\sigma} \exp(x^2)(1+\mathrm{erf}(x))dx\right]^{-1}, \tag{3.9}$$

where $V_{SS} = V_L - I/g_m$ [38, 94, 36, 19]. I used the alteration to this formula from [19], namely replacing the upper bound of the integral with $(V_{thr} - V_{SS})\,(1+0.5\tau_{AMPA}/\tau_m)+1.03\sqrt{\tau_{AMPA}/\tau_m}-0.5\tau_{AMPA}/\tau_m$ which results from the noise having a timescale of τ_{AMPA} (notice as $\tau_{AMPA} \to 0$ we recover the original bound).

This double integral results in numerical imprecision and occasionally yields NaNs (Not-a-Numbers, resultant from numerical error in the computation). A comparison of the resultant firing rates after 400 milliseconds of simulation time without input, as can be seen in Figure 3.7b, shows that the approximate fitted functions have the added benefit of being more numerically stable but not perfect. The remaining instabilities often arise from the Jahr-Stevens linearisation and in particular the computation of the effective NMDAR reversal potential (given in the Appendix equation (A.7)).

3.5.3 Truncating ψ

In [19] and [36], $\psi(\nu)$ gives the asymptotic fraction of open NMDAR ion channels under steady presynaptic firing at a rate of ν. Explicitly, the function is

$$\psi(\nu) = \frac{\nu\tau_{NMDA}}{1+\nu\tau_{NMDA}}\left(1 + \frac{1}{1+\nu\tau_{NMDA}}\sum_{n=1}^{\infty}\frac{(-\alpha\tau_{NMDA,rise})^n T_n(\nu)}{(n+1)!}\right), \tag{3.10}$$

which in turn depends on the terms

$$T_n(\nu) = \sum_{m=0}^{n}(-1)^m\binom{n}{m}\frac{\tau_{NMDA,rise}(1+\nu\tau_{NMDA})}{\tau_{NMDA,rise}(1+\nu\tau_{NMDA}) + m\tau_{NMDA,decay}}$$

To compute ψ, one needs to truncate the infinite series in (3.10) to some finite n. I chose $n = 5$, as it shows strong convergence at this value (see Figure 3.8).

3.5.4 Other Changes

For the excitatory populations, I also increased the baseline noise from external inputs towards 2mV by linear interpolation i.e.

$$\sigma_{V,\text{used}} = \sigma_{V,\text{original}}(1-\lambda) + \lambda\,2\text{mV}$$

where $\sigma^2_{V,\text{original}} = g^2_{AMPA,ext}(V_{drive} - V_E)^2 C_{ext}\tau_{AMPA}/(g^2_m\tau_m)$ was the original noise. I found this to yield more stable activity, particularly with $\lambda = 0.8$, and it may be justified by the observation that the noise was underestimated by disregarding the NMDA-driven and recurrent-AMPA-driven noise effects (see the Appendix A.2 and [19, 36]). However, I performed no further analysis of this. The noise for the inhibitory interneuron population was left unchanged.

3.6 Implementation

Algorithm 1 shows pseudocode for the full implementation implementation. In principle, parallelisation can be performed over either the evaluations of the genomes (as done in this project) or over the multiple restarts.

The spiking neural network models were written using the Brian2 simulator for the python programming language [119], while the EA was implemented using the default CMA-ES implementation in the python library DEAP [120] and with the default parameters[2] with the exception of increasing the number of offspring λ_{EA} per generation to 16, as alluded to in [21] for noisy spherical functions. The code was parallelised using Dask, and massive speed improvements were achieved using Numba's just-in-time compilation.

All simulations used the Euler-Maruyama method. For the firing rate simulations, the timestep size was 0.25ms while the spiking model simulations used Brian2's default setting of 0.1ms.

The code is available at github.com/DeanTM/masters-scripts.

[2]The parameters and their values can be found at deap.readthedocs.io/en/devel/api/algo.html#module-deap.cma.

Algorithm 1: Psuedocode for the full implementation

```
Result: hof, evolution_trajectory
cma_params ← initialise_CMA() ;
hof ← new_Hall_of_Fame(N_best);
evolution_trajectory ← new_list();
for g ← 1 to N_generations do
    Γ_g ← get_population(cma_params);
    for γ ∈ Γ_g do
        // Genotypes γ may contain weight parameters or plasticity parameters
        total_reward ← 0;
        for i ← 1 to N_restarts do
            weights, θ ← get_initial_values(γ);
            network ← initialise_network(weights, θ);
            for j ← 1 to N_trials do
                // run_RDM_trial performs the Euler-Maruyama simulation
                reward_trace, weights, θ, NaN_bool ← run_RDM_trial(network, γ);
                if NaN_bool then
                    // Penalise numerical error
                    total_reward ← total_reward - penalty;
                    break;
                else
                    total_reward ← total_reward + integrate(reward_trace);
                    // Further successive trials start from the final state of
                       prior trials
                    network ← initialise_network(weights, θ);
                end
            end
        end
        // Compute average reward
        total_reward ← total_reward / N_restarts;
        // Assign fitness to genome
        γ ←total_reward;
    end
    cma_params ← update_CMA_params(Γ_g);
    // Hall of Fame tracks N_best best performing candidates
    hof.update(Γ_g);
    evolution_trajectory.append(cma_params);
end
```

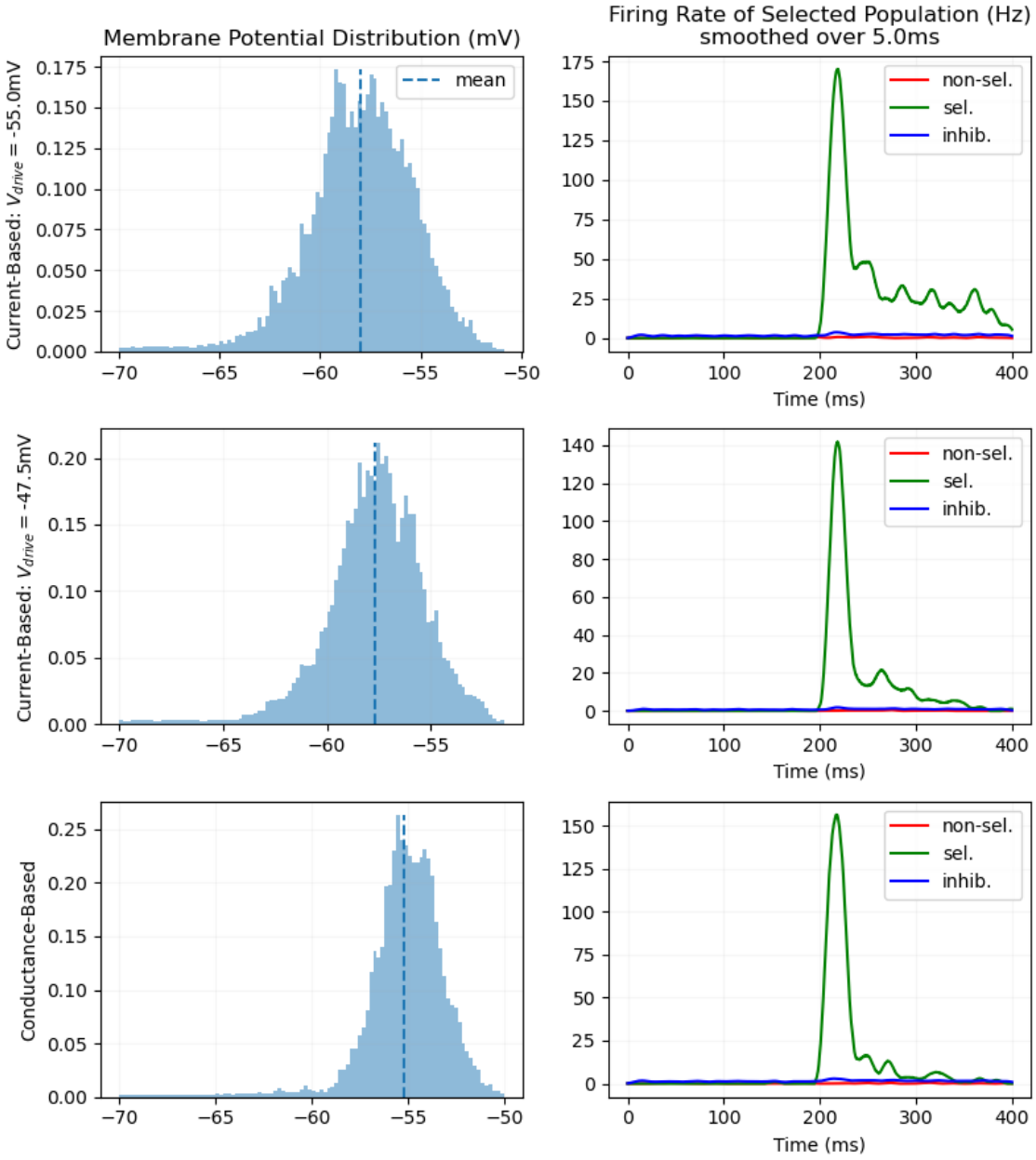


Figure 3.3: Comparison of membrane potential distributions and firing rates for two values of V_{drive} with the original conductance-based model. The left column shows histograms of the membrane potentials of the non-selective excitatory neurons (the largest population) restricted to above -70mV (or V_I) as the current-based model can escape this bound. The right column shows firing rates of three populations over the course of 400ms. At 200ms a Poisson input was provided to the selective population. Other selective populations are not shown. Code for the simulation was written using Brian2 and adapted from the Brian2 documentation at brian2.readthedocs.io [119].

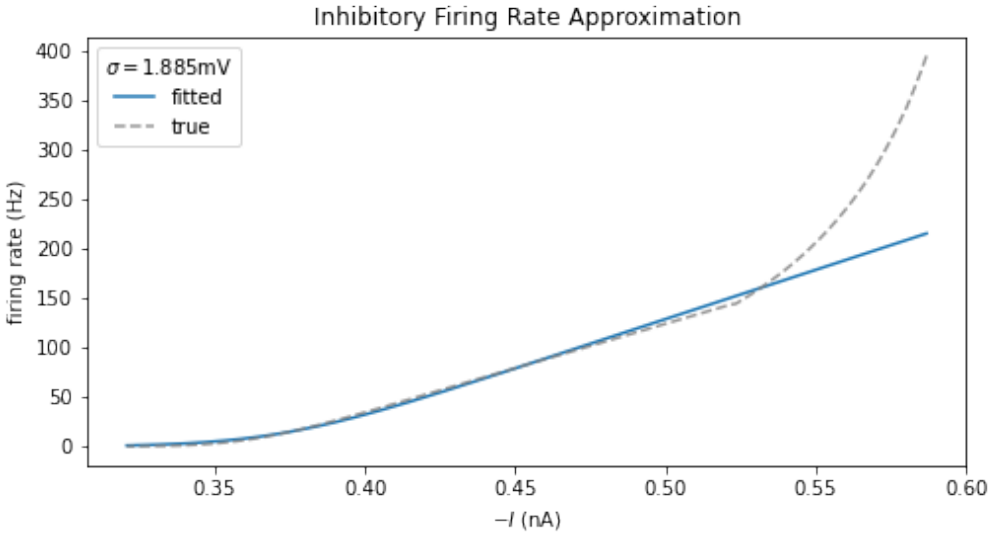


Figure 3.4: Firing rate curve approximation $\hat{\phi}(I)$ plotted as a function of $-I$ (i.e. of current entering the cell).

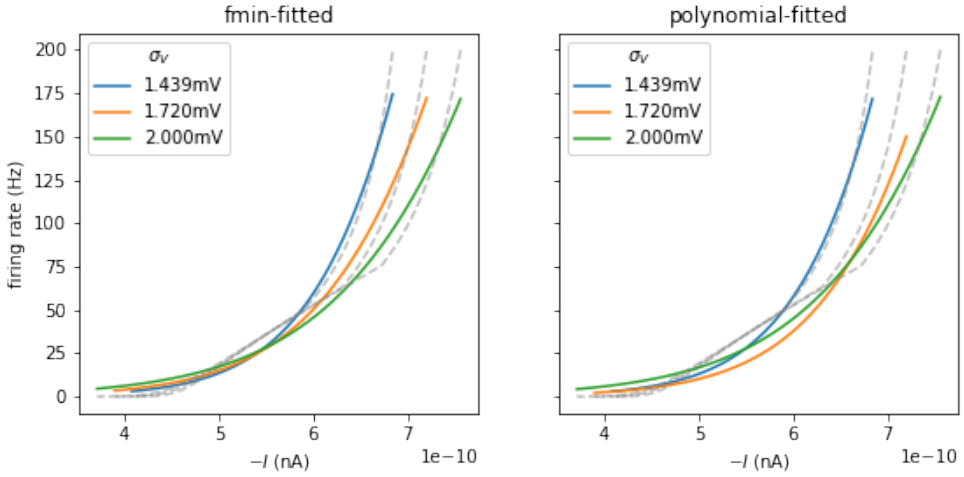


Figure 3.5: Firing rate curve approximations of the family $\hat{\phi}(I, \sigma_V)$ plotted as a function of $-I$ (i.e. of current entering the cell). The curves computed with the inverse first-passage time formula are shown in grey.

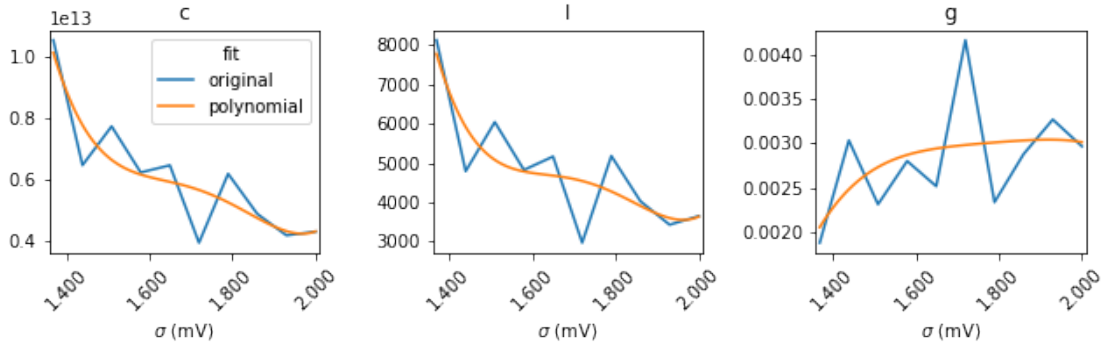
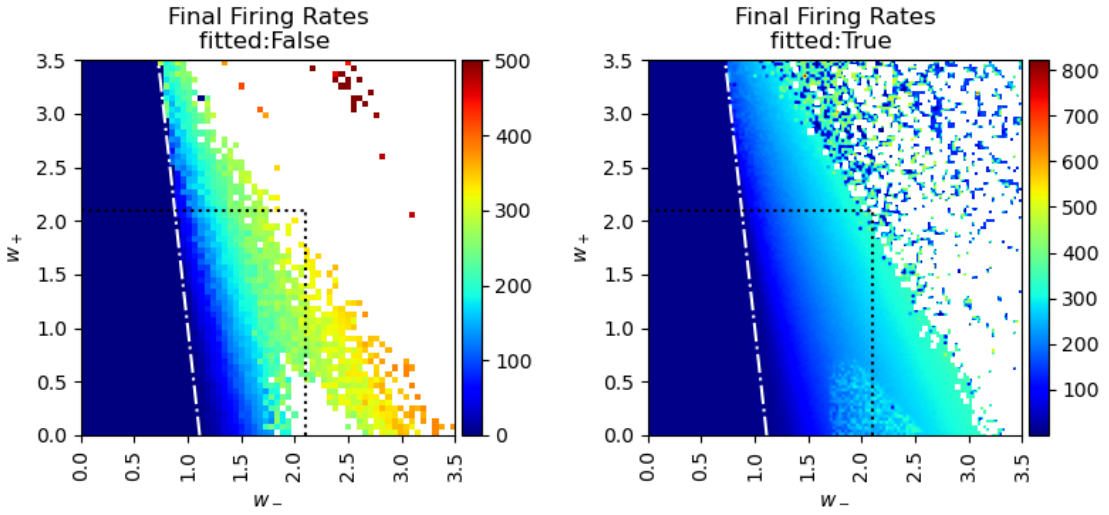


Figure 3.6: Polynomials were fitted for the functions in (3.7) by varying the noise and finding parameters which reduced the mean-squared-error between the resultant function and the true firing rate curve.



(a) Stability with Siegert formula (3.9).

(b) Stability with approximate formulae (3.6),(3.7).

Figure 3.7: Final firing rates as a function of inter- and intra-population synaptic strengths w_- and w_+, respectively. Without changing stimulus, the simulation was initialised and allowed to run for 400ms of simulation time. The maximal firing rate of any population was then recorded. Each pixel corresponds to a distinct simulation. The white pixels arise from instances where the final firing rate was NaN. The white dotted-dashed curve reflects the values used in [19], while the black box shows a projection of the region of feasible weights for the EA if we allow for the parameters used in [19] i.e. setting $w_{max} \geq 2.1$. The decrease in computation time is reflected in the fact that Figure 3.7b was several times faster (roughly 50x faster) to compute despite having 4 times as many simulations. It can be seen that using the approximate firing rate functions helps avoid numerical error, but is not sufficient to guarantee absence of errors. For more information, refer to the Discussion Chapter 5.

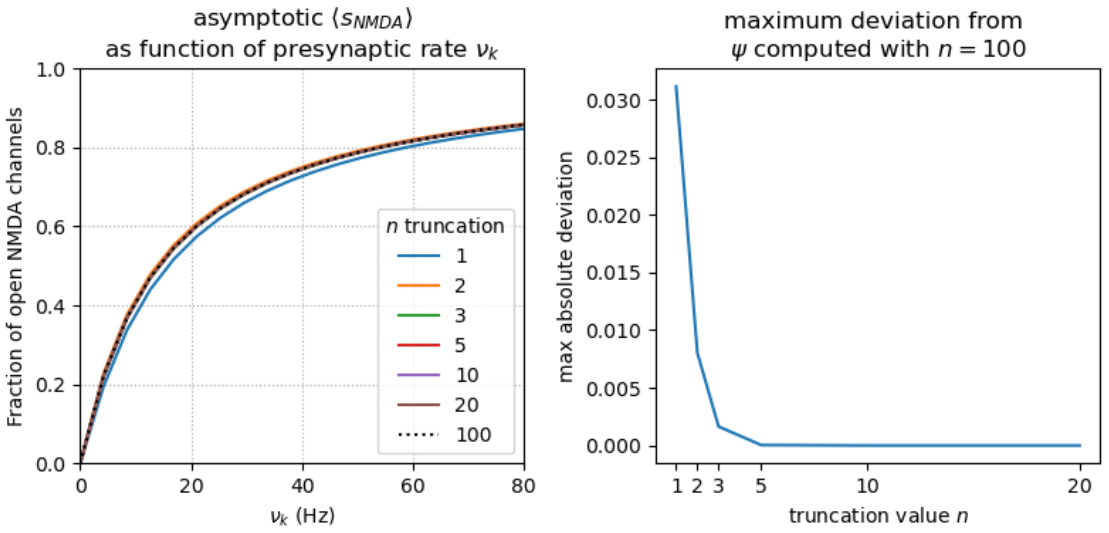


Figure 3.8: ψ, the function yielding the asymptotic fraction of open NMDAR channels, is described as an infinite series. However, this series rapidly converges. On the left are various plots for different values of n, showing that all the curves are close. On the right is the maximum deviation from curve for ψ truncated at $n = 100$ summands.

Chapter 4

Results

ἐκ Χάεος δ᾽ Ἔρεβός τε μέλαινά τε Νὺξ
ἐγένοντο: Νυκτὸς δ᾽ αὖτ᾽ Αἰθήρ τε καὶ
Ἡμέρη ἐξεγένοντο
From Chaos came forth Erebus and black Night; but of Night were born Aether and Day

Hesiod, Theogyny, line 24,
translation by Rev. J. Banks,
found at Perseus

The optimal learning rule for a stationary environment is to start with the optimal configuration and proceed to change nothing. Therefore that evolving initial synaptic weights and the plasticity rule parameters together should fail to lend any useful insights about the plasticity rule; the optimal weights will be found and the optimal rule will do nothing.

To estimate the optimal average performance for each task coherence, I evolved the synaptic weights directly using the same fitness functions but only using 10 successive trials due to limited computational resources, and because the weights did not change between successive trials.

Due to the fact that the cost of a penalty due to encountering numerical error, as well as the probability of such error occurring, grows with the number of successive trials, it follows that this change results in a slightly different fitness function - however, one that admits higher scores by having less penalisation. As such, it can still be used as an upper bound on the achievable scores with 100 successive trials by multiplying the score by 10.

In what follows, I discuss evolving the weights directly, then evolving the learning rule, and finally address the question of whether the same learning rule was evolved for each coherence value.

This was not the case.

4.1 Evolving The Weights

The CMA-ES EA was used in an attempt to evolve weighs both on the RDM task and on the XOR task. For the RDM task this was to determine optimal potential performance of the rate-based model, while for the XOR task this was to test the expressivity of the rate-based model.

4.1.1 The RDM Task

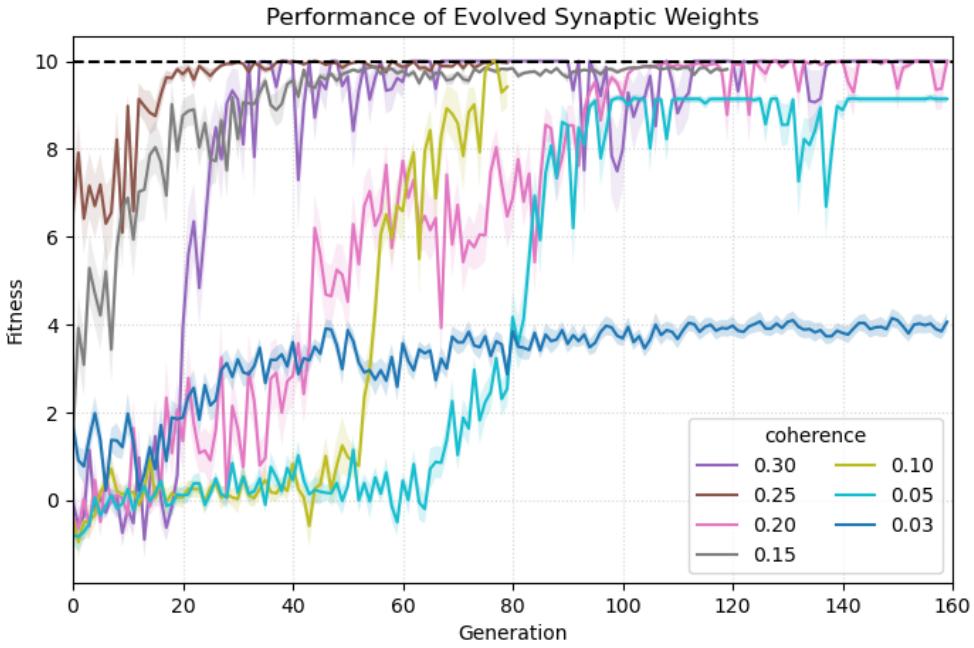


Figure 4.1: Attempts to solve for the optimal synaptic weights were run with different initial conditions and parameters. A range of initial synaptic strengths with recurrent synaptic strength $w_+ \in [1, 2.1]$ and interpopulation excitatory strengths $w_- = 1 - f(w_+ - 1)/(1 - f)$ were used, alongside varying initial noise strengths for initial isotropic Gaussian mutations Σ_0 in the range of 0.01 and 1, and varying maximal numbers of generations in the range of 40 to 200. This figure shows the trajectories of the evolution for each coherence value which achieved the maximum population average minus one population standard deviation. The maximal possible score of 10 (as 10 successive trials were used) is shown by the dashed line. One fifth of a standard deviation is shaded around the population trajectories for consistency with Figure 4.2.

When evolving the weights, the aim was to determine the optimal possible weights against which to compare the plasticity rules. Thus various restarts were used, sampling initial weights from the curve $w_- = 1 - f(w_+ - 1)/(1 - f)$ with w_+ between 1 and 2.1, where $f = 0.1$ is the fraction of excitatory neurons in each selective subpopulation (all parameters are included in the Appendix Section A.3). Various levels of noise were used as well, alongside various maximal numbers of generations. Some evolutionary trajectories converged after under 40 generations (such as for

coherence of 0.25) while others took over 100 generations (such as for coherence 0.30), reflecting the stochasticity of evolutionary algorithms. What is shown in Figure 4.1 as well as in 4.3 are only the trajectories which achieved the maximum population average minus one standard deviation for each coherence i.e. the trajectories which maximised:

$$\max_g \left[\langle \Phi(\gamma) \rangle_{\Gamma_g} - \sqrt{\left\langle \Phi(\gamma) - \langle \Phi(\gamma) \rangle_{\Gamma_g}^2 \right\rangle_{\Gamma_g}} \right]$$

where Φ is the average performance over 10 successive trials averaged over 10 repeats.

Because optimal possible performance of 10 was achieved for lower coherence values, weights were not evolved for the higher coherence values.

4.1.2 The XOR Task

The CMA-ES algorithm was unable to successfully solve the task, and optimal performance remained near zero for all values of p and all strengths of stimuli. This was taken as an indication that the model could only solve simpler problems, and no further investigations of performance or evolution on the XOR task were made.

4.2 Evolving the Learning Rules

The first thing one might notice here is that the weights evolved more slowly compared to the learning rules, in terms of number of generations. This is likely an artefact both of the different space from which the genomes were sampled, with different scales, and of potentially reduced noise due to averaging effects from the increased number of sequential trials when evolving the learning rule.

4.2.1 Were The Same Rules Evolved?

The learning rules were all evolved to fit a slightly different version of the same task. One might ask, is the learning rule the same in each case? Comparing the parameters of the learning rule in isolation would mean very little here: the evolutionary algorithm determined the learning rule conditioned on the stimulus and neural activity. Two distinct learning rules may differ - even radically - on some given inputs, but if those inputs arise with negligible probability on the underlying task then their difference will not be reflected in the fitness of the evolutionary algorithm.

As such, to determine whether two learning rules are different we need to determine how they behave on the same task. This is limited because as Figure 4.3 shows the performance for different coherence levels differs greatly and so we cannot pick one representative coherence

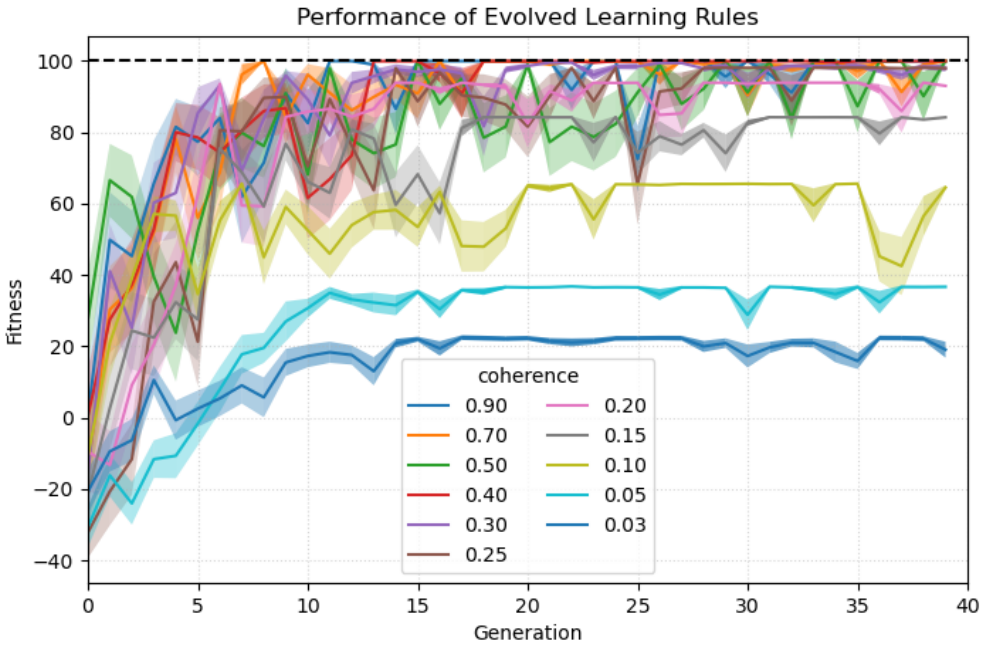


Figure 4.2: Average fitness scores per generation of the CMA-ES algorithm. For clarity the shaded regions show one-fifth of a standard deviation determined by the individuals of that generation. Above a coherence level of around 25% the model is able to achieve near perfect performance despite the numerical instabilities. See also Figure 4.3.

level. Nonetheless, should we pick one, it makes sense to pick the one which may yield the most information. To this end, we can choose the coherence level which maximised the empirical entropy H_{emp} of the best candidate's performance.[1] If we denote each candidate genome by γ_c where c is the coherence of the task, and the fitness distribution of candidate γ_c on its task with this coherence as $f_{dist}(\gamma_c, c)$, then we take

$$c_{test} = \text{argmax}_c H_{emp}(f_{dist}(\gamma_c, c))$$

The entropy is approximated from the histograms of the samples used in creating Figure 4.3 with 20 bins within the range of -100 and 100 (the minimum and maximum possible scores). These entropies can be seen in Figure 4.4 and as can be seen, the lowest tested coherence of 0.03 yielded the learning rule with the highest entropy. Continuing with our scheme, we use $c_{test} = 0.03$.

Having chosen the coherence level c_{test}, we can evaluate all the candidates (one evolved for each coherence) on the task with coherence c_{test} to obtain distributions of their performances $f_{dist}(\gamma_c, c_{test})$. This gives us a means to compare the distributions of performances on tasks as a proxy for comparing the learning rules themselves.

But how do we compare them? We could empirically compute pairwise Kullback-Leibler divergences, but this would yield scores difficult to determine. Instead we can use a non-parametric

[1] This is in fact the strategy which motivates the use of normal distributions in evolution strategies, as one wishes to maximise the information gained with each new offspring [99].

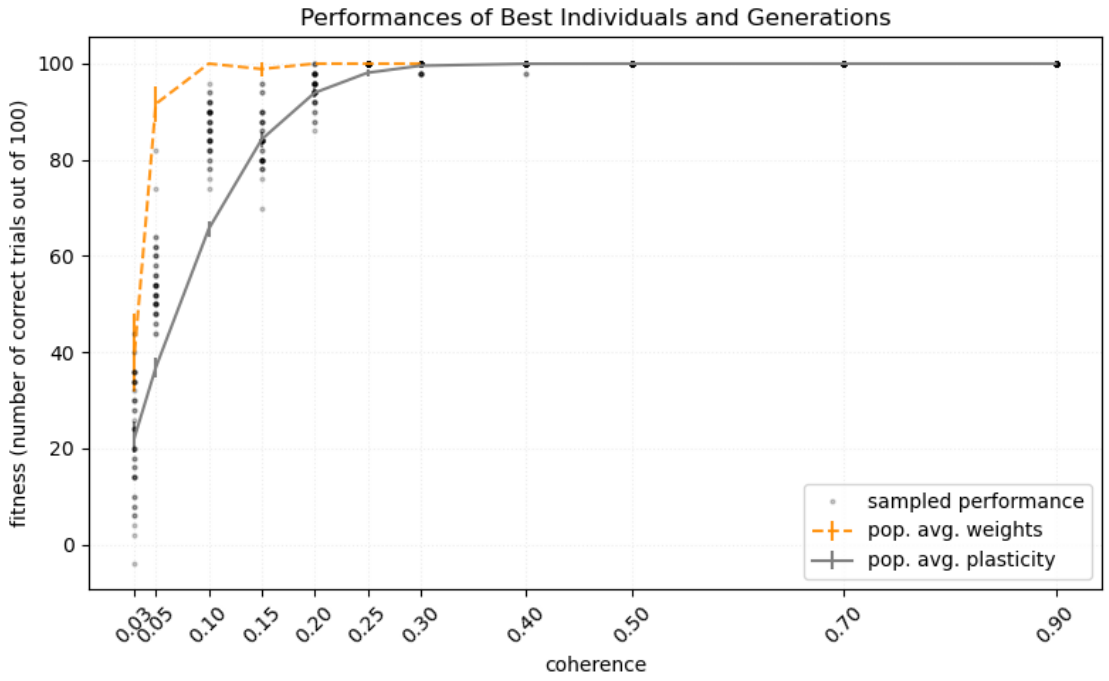


Figure 4.3: Maximal performance for each coherence level. The grey line shows the average *population* performance for the generation which highest average minus one standard deviation score for evolving the plasticity rules. The orange line shows the same, but for evolving the weights directly, with fitness and standard deviation scaled up. Dots represent independent performances of the best overall learning rule candidate, without averaging over multiple trials. The fact that perfect performance is achieved repeatedly for the higher coherence values suggests that the task can be solved without any plasticity at all, given the initial conditions of the network.

two-sample test such as the Kolmogorov-Smirnov test. In order to reduce the number of tests and avoid artefacts of multiple testing, we can use one-versus-rest two-sample tests. To this end we need to use a Bonferroni or Holm-Bonferroni correction. Nonetheless, not all comparisons are created equal, as the performance for learning rules fitted to tasks with coherence scores closer to c_{test} may be more similar than those fitted to tasks which are more dissimilar. The next question we need to consider is if *any* learning rule is different, or which learning rules are different i.e. do we want to correct for family-wise error rate, or for individual comparisons? The choice here is important, but a choice made prior to the evaluation. It would be statistically unsound to perform both tests with the same data where, in our case, the data includes the fitted parameters.

One can suspect that the case of the higher coherence tests, the task is trivially simple enough that very little learning is required. If this is the case, then the corresponding learning rules conceptually arose from little more than a random walk on a fairly flat landscape and are likely to diverge in their behaviour from the others. In order to know how such task difficulty, or coherence c, might translate to deviations in performance distributions $f_{dist}(\gamma_c, c_{test})$ it will be helpful to consider each test individually and not focus on diminishing the family-wise error rate. Hence, the Bonferroni correction is used.[2]

[2]A small point should be made here. The Bonferroni correction assumes that the tests are independent. In this case they are not, as we are comparing different samples to overlapping control samples. As such, a test such

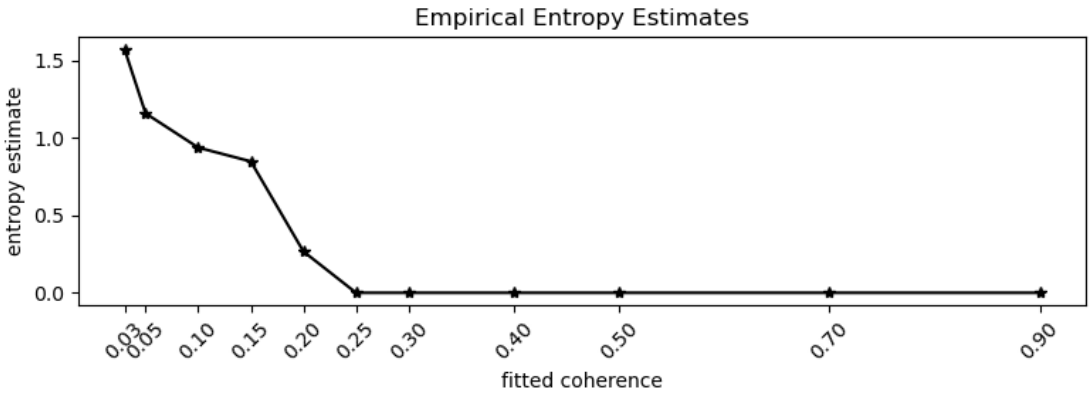


Figure 4.4: Entropy estimates of $f_{dist}(\gamma_c, c)$, the performance distributions of the best learning rule candidates, as a function of coherence c.

As we can see in Figure 4.5, each distribution $f_{dist}(\gamma_c, c_{test})$ can be seen as distinct: we can reject the null hypothesis for any of the selected learning rules that the samples of its fitness distribution on the RDM task with coherence $c_{test} = 0.03$ arose from the same distribution as the distribution formed by the aggregate of the remaining rules' data. One needs to be careful here: one outlier sample in this test could allow us to reject all the null-hypotheses. But the point is made, that at least one learning rule is sufficiently different that the aggregate behaviour can be distinguished from any individual rule.

as Dunnett's test should be used. However, due to the possibility of encountering numerical errors and receiving a penalty, the distributions of performance might best be written as a mixture distribution over the performance when numerical errors are not met (weighted by the probability of no numerical error) and the performance when numerical errors are met (weighted by the probability of an error in one of the sequential trials). This mixture distribution would not satisfy the normality assumption for a parametric Dunnett's test. Thus the Bonferroni correction seems to be a fair compromise.

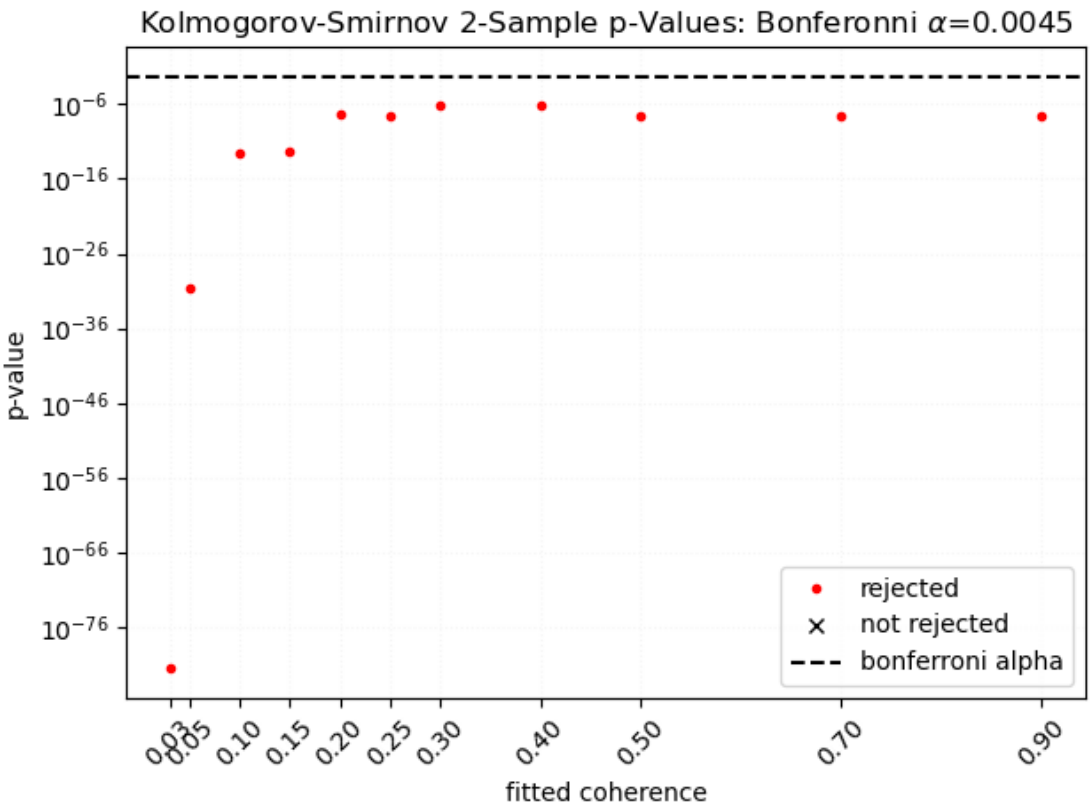


Figure 4.5: p-Values of the Kolmogorov-Smirnov significance tests. As can be seen, the null hypothesis that the fitness scores are generated by the same underlying learning rules can be rejected in every case. What we observe is that each evolved learning rule is different.

Chapter 5

Discussion

So much universe, and so little time.

Sir Terry Pratchett

There remain many more avenues to explore, extensions which can be made to the Wang model, different RL tasks to try and different optimisation procedures to use. In this section I hope to address the questions and limitations of the procedure implemented here, and propose further steps which might be taken.

In summary, we started with a spiking recurrent neural circuit decision making model, a family of STDP plasticity rules extended to allow for reward-driven plasticity, a RL perceptual decision-making task and an understanding that normative or optimisation-guided approaches may lead to insight in biology.

However, for computational expediency we reduced the spiking model to a rate-based model, and the STDP framework to a rate-based framework. Any results arising from this reduction can only be accurate insofar as the reduction is a faithful representation of the original model. This leaves open questions about the original model, as well as the original model's relationship with biology, all of which I aim to address here.

5.1 Was The Reduction Necessary?

Early in this project I initially hoped to evolve STDP rules directly with a biophysically inspired SNN. Such a model might be able to perform p-ary decision making tasks by adjusting the parameter p, or continuous decision making tasks by implementing a bump attractor on a ring, as has been used in [79, 80, 81]. The pure computational load dashed these hopes. When evolving

an individual learning rule over 40 generations, using 100 sequential trials with 10 restarts and a population of 16 candidates, without encountering any numerical errors this amounts to 640000 individual simulations. The SNN simulation, when incorporating interaction with the environment, took about 30 minutes per simulation. Even distributed optimally over 16 cores this would take over 2 years on the hardware available to me. And this back-of-the-envelope calculations fails to include that the STDP rules had more free parameters, amounting to a higher-dimensional search space for the EA and thus potentially requiring a larger population and more generations before convergence. While my SNN was by no means optimised[1] the scale of the problem is still apparent.

However, it is also the case that the family of learning rules is too inclusive. As is shown in Chapter 3, the many parameters are redundant and even so, different learning rules were found to solve the task at different coherence levels. Thus a prudent first step would be to incorporate biological considerations to further reduce the search space. This may come from limiting the range of the time constants (eligibility traces for dopamine-driven learning appear to be on the order of 1 second [3], while the time constants of the Volterra kernels are likely determined by underlying synaptic mechanisms such as the NMDA activity), or setting some of the Volterra kernels such as $\mathcal{F}_2^{X,X,Y}$ to zero so as to disallow, for example, pre-pre-post interactions, as these are not needed for a good fit to data [23].

If an attempt is made to decouple Hebbian or unsupervised and reward-driven plasticity, then speedups may also come from separating unsupervised and reward-driven trials where the unsupervised trials can be simulated significantly faster as no action readout is necessary. This may capture learning on a task where the presence of a reward signal may not occur at all with some probability.

All in all, it is possible that with further work the EA may be implemented on the SNN directly.

5.2 Rectifying the Model

Several features of the network model, the task and the plasticity rules may be eligible for optimisation of tweaking. These features are discussed here.

5.2.1 The Network Model

Initially I attempted to implement a conductance-based mean-field model as discussed in the Appendix Section A.2. However, due to the self-consistent calculation of the effective parameters with the average membrane potential and firing rate, this lead to a much slower simulation than

[1] The root cause of the slow simulations seemed to be my use of Brian2's `network_operation` function to determine action selection and reward signal, which would not be necessary on an unsupervised task

the original spiking model.[2] This does not imply that the implementation is necessarily slow; models such as in [122] and [123] suggest that fast conductance-based mean-field simulations are very feasible.

The simple act of changing the model to a conductance-based model implies a need for the fitted firing rate functions $\hat{\phi}$ to depend on the effective membrane time constant, which in turn increases the number of variables in the polynomial fit. This may be doable and is unlikely to provide any significant slowdown of the simulations.

However, as the nonlinearity and bistability of the model depends on the dynamics of s_{NMDA} and at the population level its average $\langle s_{NMDA} \rangle$, it is not clear that a conductance-based model would yield qualitatively different results. The use of a conductance-based model should instead be justified by the data that the model is attempting to fit.

A few parameters were chosen without any explicit optimisation or fitting procedure, namely λ and the arbitrary noise level 2mV, V_{drive}, and the strength of the input noise $\sigma_{noise} = 0.007$nA. The degree of the polynomials in fitting the terms for the excitatory rate function approximation $\hat{\phi}(I, \sigma)$ was also arbitrarily chosen. In future, the following changes might allow the rate-based model to more faithfully capture the SNN:

1. Using an optimisation procedure to choose V_{drive} so as to minimise a weighted average of the Kullback-Leibler divergence between the conductance-based and current-based membrane potential distributions and the mean-squared error of the deviations of the firing rates in the spiking models. Due to avoiding effects of stochasticity, this procedure should use the same Poisson inputs for both the conductance-based and current-based simulations

2. The noise in the rate-based model might be better chosen to have the effective current noise strength $\sigma_C^{eff}(\langle V \rangle)$ or $\sigma_C^{eff}(V_{drive})$ of equation (A.9)

3. Considering various degrees of polynomials and comparing the mean-squared error between the "true" firing rate ϕ and the approximation $\hat{\phi}$. As the parameters of the polynomial are chosen to minimise this loss, one would expect the loss to be monotonically decreasing in degree of the polynomials. Because minimising mean-squared error is formally equivalent to maximising likelihood with Gaussian noise, one could then use the Akaike Information Criterion (AIC) to determine the degree of the polynomials [124].

An investigation as to why the excitatory noise, but not the inhibitory noise, needed to be amplified towards 2mV might also prove insightful.

[2] I suspect I made a mistake with my implementation attempts, but due to the slow simulation time this made debugging difficult. Had I the presence of mind at the time, I would have initially implemented the current-based solution and moved it to a conductance-based model by adapting a parameter as done in [121] but at the population level. I would advise this approach for future prospects the reader may have.

Alternatively, to consider a more numerically stable model, it might be better to drop the voltage dependence for the NMDAR ion channels. The Jahr-Stevens linearisation around the average membrane potential $\langle V \rangle$ yields the description of the effective NMDA reversal potential $V_E^{eff}(\langle V \rangle)$ and effective NMDA conductance $g_{NMDA}^{eff}(\langle V \rangle)$ in equations (A.7) and (A.6). However, in the expression of $g_{NMDA}^{eff}(\langle V \rangle)$ we have the term $\frac{1}{J_2(\langle V \rangle)}$ where for $\langle V \rangle \approx -27\text{mV}$ we have $J_2(\langle V \rangle) = 0$. This division-by-zero leads to numerical errors and indeed while the average membrane potential $\langle V \rangle$ as calculated by (A.15) can reach such high values, in the SNN no membrane potential can exceed $V_{thres} = -50\text{mV}$ (for reference, see Figure A.3). Rectifying $\langle V \rangle$ to remain below -50mV or disregarding the Jahr-Stevens formula entirely and relying only on the nonlinearity of $\langle s_{NMDA} \rangle$ might yield more stable code. This has yet to be explored, but the recurrent neural circuit model requires at least some nonlinearity for the stability of increased population activity.

Interestingly one might also find improvements if they use an EA to determine the network dynamics directly. In [125] the question of using an evolutionary algorithm to determine a differential equation was investigated. They used DE to determine coefficients for a polynomial approximation to differential equations and determined that the absolute error was equal (when zero) or less than that of the Runge-Kutta Nystrom method, and that it required less CPU time to integrate the dynamics.

5.2.2 The RDM Task

The RDM task itself is also very simple. Recurrent neural circuit models have been included in foraging tasks [24] and can easily be extended to many-option decision making tasks such as k-armed bandits [82]. These tasks can be designed so that the environment changes on multiple timescales, which may lead to undermatching in the learned behaviour [85]. Moreover, if the environment changes, there may be no unique solution for the weights which maximises reward and as such the synaptic weights and the plasticity rule may be determined in concert. However, due to time constraints I only succeeded in fitting plasticity rules and independently synaptic weights for the RDM task. Nonetheless, this constitutes a proof of concept.

After determining that a different plasticity rule was fitted for at least some of the coherence values, one might ask why I did not determine a single plasticity rule for all of the coherence values? The question then becomes, how do we fairly evaluate the fitness of a plasticity rule with multiple coherence values? There are a few ways one could go about this, but many of them are in some way dubious.

If each coherence value is tested evenly, then the plasticity rule would be biased towards solving the solvable problems. We can imagine a process whereby one third of the trials used a coherence of 0.01 while another third used 0.99 and the last third used 0.15. Since the scores with the lowest coherence are unlikely to improve much with plasticity, while the scores with the highest coherence

are likely to be good without plasticity, the plasticity rule would focus on improving on trials with the last coherence value. Whether this is appropriate behaviour or not for a plasticity rule is unclear. If we wish to try a weighted average of the scores on trials with differing coherence values, the weightings would need to be determined. If we used the harmonic mean over scores with various coherence values, then the inclusion of a single unsolvable task would in the limit reduce the harmonic mean to zero thus rendering the fitness function constant.

If the differing coherence values are presented in a fixed sequence, then the plasticity rules may evolve to optimise for the sequence every bit as much as they did, in this implementation, evolve to be optimised for individual coherence values. Alternatively, if the differing coherence values are sampled randomly then an extra source of noise is added to the evolutionary algorithm which is difficult to mitigate: how do we decouple which individual performed best from which individual, by chance, was given the easiest trials? The best learning rules, after all, might be the ones which perform best on the trials of *intermediate* difficulty.

One solution might be to consider a boosting scheme such as used in boosting of weak learning algorithms [126]. One could iteratively reweight the individual coherence trials on successive restarts of the full CMA-ES in accordance with prior performance. However, at best this would be immensely time-consuming.

All in all, I believe the lesson here is that the fitness function needs to be carefully considered.

5.2.3 The Plasticity

Two different versions of synaptic traces were used and implemented in two separate ways, simultaneously. This should not have been done, but the mistake only became apparent at the later stages of writing this thesis.

On the one hand, the convolutions with exponential decay Volterra kernels such as $\mathcal{F}^{X,Y}(s) = F^{X,Y}\exp\left(\frac{-1}{\tau_{F,X,Y,0}}\right)$ in (2.14) (omitting the later dependence on θ and w for clarity) can be implemented online with "synaptic tags" $z_{F,X,Y,0}$ following the dynamics [16]

$$\tau_{F,X,Y,0}\frac{dz_{F,X,Y,0}(t)}{dt} = -z_{F,X,Y,0}(t) + F^{X,Y}Y(t) \tag{5.1}$$

The average value of such a tag is $\langle z_{F,X,Y,0}(t)\rangle_T = \left\langle \int \mathcal{F}^{X,Y}(s)Y(t-s)ds\right\rangle_T = \tau_{F,X,Y,0}F^{X,Y}\langle Y\rangle_T$ where $\langle Y\rangle_T$ is determined over a time $T > \tau_{F,X,Y,0}$, and hence I used the reduction

$$\left\langle \int \mathcal{F}^{X,Y}(s)Y(t-s)ds\right\rangle_T = \tau_{F,X,Y,0}F^{X,Y}\nu_i \tag{5.2}$$

as in [22]. Usually $\tau_{F,X,Y,0}$ would be of the order of tens to hundreds of milliseconds (such as in [22]). If ν_i is changing on a timescale faster than $\tau_{F,X,Y,0}$ then this approximation becomes

inaccurate.

However, the fractions of open synaptic channels $\langle s_{AMPA} \rangle$ and $\langle s_{GABA} \rangle$ follow the same dynamics, but on timescales of 2ms and 10ms respectively, yet are not implemented as $\langle s_{AMPA/GABA} \rangle = \nu_j \tau_{AMPA/GABA}$. As a first step to resolving this discrepancy, one might use this implementation, but the absence of synaptic traces, or tags, for the plasticity rules needs to be given further thought. We also need to consider that the firing rate itself seems to change on a timescale of $\tau_r = 2$ms, at least when implementing a step current.

Firstly, if one were to interpret the phenomenological features, it is likely that the tags correspond to some trace of NMDAR activity as the effects of neuromodulated activity was determined to depend on the NMDARs [3, 127]. Hence it might be feasible to consider learning rules which use exactly the fraction of open NMDAR channels s_{NMDA} (or their average $\langle s_{NMDA} \rangle$ for the rate-based model) as their synaptic tags. I avoided that here as I explicitly wished to avoid presuming a substrate for the phenomenological rules.

Ignoring any biological interpretation of the phenomenology, we still need to interrogate the implications of the approximation (5.2). Without loss of generality, I will discuss only the kernel $\mathcal{F}^{X,Y}$ with decay time $\tau_{F,X,Y,0}$ and the postsynaptic population rate ν_i, but the same considerations apply to the remaining kernels and the presynaptic population rate ν_j. Using the approximation allows us to reduce the number of free parameters for the evolutionary algorithm as discussed in Section 3.4, as implementing individual synaptic tags would require one tag for every decay time (such as $\tau_{F,X,Y,0}$ and $\tau_{G,X,Y,0}$ which are brought together into one parameter $\bar{\xi}_0^{11}$). But this comes at the cost of assuming that the instantaneous firing rates ν_i are good approximations of the average firing rates over the prior time $T > \tau_{F,X,Y,0}$, that is, roughly the time which the synaptic tag $z_{F,X,Y,0}$ would take to converge.

As a first attempt to justify this one might appeal to ergodicity, but conceptually even though the full system may be ergodic, due to the attractors the time it takes for the activity to escape the basins of attraction by chance effects alone is large. This is further exacerbated by the fact that the noise driving the fluctuations in firing rate is coloured Ornstein-Uhlenbeck noise, so that large fluctuations become exceedingly rare. The system may be ergodic in the sense that we can interchange trial-averages (as thus population averages, assuming independence of the neurons and synapses) with long time averages, but that time needs to be at least comparable to the time required to escape the basins of attraction which is longer than an individual trial in most cases.

On the other hand, when the activity is near an attracting point the activity is fairly constant. Thus, one might argue that it is only during the transient that the firing rate varies and it is only during this brief time that the approximation is incorrect. However, this argument has two flaws. Firstly, the attracting states of elevated activity do not exist for the full range of synaptic weights [94] and thus the transient activity may be the only meaningful change in activity that occurs

during a trial. Secondly, it is during this transient that the decision is made, as the final choice is strongly determined by the trajectory towards one of the attracting states. Thus it is exactly the dynamics of this transient which needs to be adapted with the plasticity rule.

In the end, I have been unable to reconcile these differences in the synaptic traces. This remains important yet intriguing future work.

5.3 Further Directions and Generalisations

Here I will discuss some changes one might make to extend upon this work, or generalise it to more complex and biophysically realistic models.

5.3.1 Further Steps

These results might lend themselves to further immediate investigation; however time constraints prevented such investigation. The CMA-ES, through its covariance matrix, yields a Gaussian distribution of predictions for high fitness candidates; it would be interesting to determine whether the gradient of this distribution correlates with the average gradients of the fitness function in the vicinity of the optimal found candidates, as might be expected [100] and is suggested by the optimality principle (replacing natural evolution with the EA).

Although different learning rules are found for different coherence values, two more immediate questions can be asked from these results:

1. Do learning rules found for the same coherence on different runs of the EA behave similarly? Particularly if the parameters found are themselves different. This would tell us whether different parameters yield the same learning rule and, hence, suggest whether the learning rule has redundant parameters.

2. Are there regions of the search space wherein the trends of different runs of the EA agree across coherence values? While the final resultant learning rules disagree - both in terms of parameters and resultant behaviour - it may be that some learning rule parameters are better than others across all tested coherence values and this may be reflected in the drift of the populations of runs of the EA.

The next step one might do is to return to the STDP rule and test it. While it was not computationally feasible to evolve the STDP directly, our goal is still to understand plasticity in a spiking model. However, for each rate-based plasticity rule there exists a family of A-A STDP rules which, under Poissonian independence assumptions, are equivalent. An important next step would

be to determine if the different rules in the same family yield equivalent performance, possibly by again using a Kolmogorov-Smirnov two-sample test. If not, this would expose an error in our reduction assumptions which could encourage further research into more realistic rate-based approximations.

In a similar vein, it would be interesting to see to what extent the EA overfitted the rate-based model by comparing the evolution of its performance with the performance of equivalent STDP rules. One could imagine that the STDP rule improves in performance to a point, and then decreases in a manner reminiscent of the bias-variance trade-off.

We can also compare the learning rule in light of optimal behaviour. In [18] the optimal parameters are determined for maximising reward rate for decision making tasks of various difficulty in the free response paradigm. These parameters yield distributions of RTs and accuracies which can be compared to those generated by the model before and after learning on a similar task. We can ask, "Does an evolved plasticity rule adapt the RT distribution and accuracy towards that of the optimal behaviour?" Since the optimal solution is given by a linear DD model [18], we could also ask of the evolved weights, "Do the dynamics of the model with the fitted weights appear linear as in the optimal DD model?"

To find an optimal STDP rule - possibly on a more complex task which depends on specific spike times - one may ultimately determine it directly with an EA. Good initial conditions for the EA would speed up this process, and these initial conditions may be determined by the rate-based method used here. Whether or not the final rate-based rule provides good initial conditions for evolving an STDP rule remains to be tested.

Finally, one might ask why the average performance across different coherence levels was not considered. This is because such averaging would require a prior assumption of the distribution across coherence levels, but I am unaware of any normative or descriptive a priori justification for any particular distribution, including the uniform distribution.[3] Without the means to compute the weights for this weighted average, no weighted average was computed. That said, the best learning rule should in some sense be the best across multiple coherence levels. A natural extension to this work would then be to consider the dependence of the learning on different distributions over coherence levels, perhaps sampled from a beta distribution with varying parameters (of which the uniform distribution is a special case).

5.3.2 Including More Precise Biophysical Dynamics

The AdEx model of [42, 43] may better capture the subthreshold membrane potential traces following a prescribed fitting procedure and thus would allow for a voltage-dependent plasticity

[3] Indeed each simulation can be seen as an average with respect to the Dirac distribution centered at the corresponding coherence level.

rule such as the Clopath model [53, 52] (see [128] for a comparison of model fits).

The dynamics at the synapse can also be enriched. Under the classical STDP rule, the ability to form Hebbian assemblies is strongly tied to the axonal and dendritic delays [64]. If an axonal delay of τ_a and a dendritic delay of τ_d are included, then two neurons can both "perceive" themselves as being the first to fire an AP if they fire within $t_a - t_d$ of each other, which allows pre-before-post LTP to occur in both directions. The model might be extended to include shared or randomly distributed synaptic delays.

Individual synapses can also be facilitating or depressing [41, 16]. Facilitating synapses initially produce successively increasing PSCs when the presynaptic neuron repeatedly fires within a short time, before saturating or depressing, while depressing synapses produce decreasing PSCs. Including these dynamics can produce a range of effects, from short-term sustained activity with recurrent facilitating synapses, while depressing synapses can reduce the chance of the postsynaptic neuron firing whenever the presynaptic neuron fires. If the PSPs drive the plasticity rule such as with the Clopath model, depressing synapses might also regulate plasticity.

Plasticity can also interact with the dynamic synapses in another way. We can separate the synaptic strength into pre- and postsynaptic variables, such as release probability per site P_{pre}, number of release sites N_{pre}, and the charge deposited per release of neurotransmitter quantal q_{post}. Doing so allows us to unify STP models such as the Abbott model [41, 16], which models the dynamics of P_{pre} on a short time scale, with location dependent long-term synaptic plasticity models which model LTP and LTD as distributing across the pre- and postsynaptic variables depending on the current state of the variables [45, 46, 47]. This can yield benefits on certain tasks, such as reducing the time it takes for a network to relearn a task after being trained to perform another task [45], or help to maintain a balance of excitation and inhibition or provide a fit to more experimental data [47]. Due to these benefits, we should be able to investigate these rules with an optimality-first normative approach.

Another class of plasticity rules may be included as well. Inhibitory plasticity, or plasticity of the inhibitory synapses, is much less studied but seems to perform a different role of maintaining E/I balance and keeping the network in an asynchronous firing regime [48]. The dynamics for the inhibitory plasticity are most likely different to those of the excitatory synapses. In this thesis I have only included plasticity of excitatory synapses, but one might simultaneously search for an inhibitory plasticity rule as in [25]. At the time of writing this, I am not aware of any attempt made to combine inhibitory plasticity rules such as Inhibitory STDP (iSTDP) with a recurrent neural circuit model, but it does pose the problem that the maintained increased activity of the selective populations may have their activity downregulated by the adapting inhibitory inputs, as indeed this is what happened when I initially attempted to combine iSTDP with the Wang model.

Having implemented one of these changes at the level of the SNN, it raises the question of

whether mean-field theory can still be used to find a faithful population-level model. The mean-field theory of conductance-based neurons has recently been extended to include dynamic synapses [123]. This impressive work builds upon the AdEx model and captures elevated sustained activity.

Alternatively, using quasirenewal theory [122] derive a model which captures the mesoscopic effects of a neural population, such as oscillations in response to a step current (see Figure 5.1). The model used here, on the other hand, will only show transients of a steady increase or decrease to the new asymptotic firing rate in response to application or removal (respectively) of a step current input. Although built for current-based synapses, the model of [122] can be adapted to use conductance-based currents and dynamic synapses.

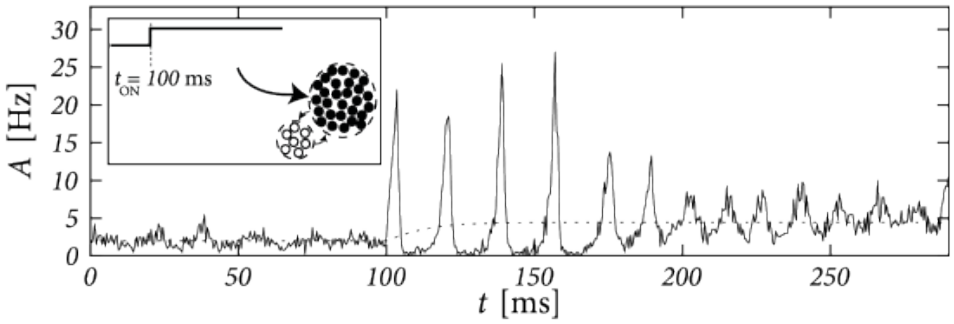


Figure 5.1: The transient population activity (denoted here by A) of a population of LIF neurons in response to a step current at 100ms. The dotted line shows the dynamics of a rate-based model as considered here. Recent work, such as [122], can capture the oscillations in a rate-based model. Image taken from [38], at neuronaldynamics.epfl.ch/online/Ch15.html.

From a different angle, rather than using a mean-field model one might consider evolving the Fokker-Planck dynamics (for the membrane potential distribution) directly. In [129] it is explained that this can be done rather efficiently by considering the eigenbasis of the Fokker-Planck operator. Importantly, most modes will decay rapidly, requiring only a few of the eigenfunctions to be considered. The firing rate is then obtained by the flux of the membrane potential across the threshold. It is speculative, but knowledge of the membrane potential distributions may allow one to consider higher order moments of the synaptic plasticity dynamics.

5.3.3 Easier to Fit Models

One of the motivations for the procedure implemented in this thesis would be to derive a prior distribution for parameters of a model so as to implement an optimality-driven regularisation model fit. If one wants to fit the model and the learning rule simultaneously to data, this is only helpful if the model itself can be easily and well fit to the data, and so it may be prudent to choose such a model.

The previous subsection discussed ways to make the model more accurately fit biological data, and in particular the AdEx model can be fitted with a prescribed procedure. In the less biophysical

direction, the linear SRM with escape noise can be fitted to data by determining the convolutional kernels, while the Linear-Nonlinear Poisson (LNP) model is a simple example of a Generalised Linear Model and as such maximum likelihood estimation with this model is a convex optimisation procedure [38]: any local optima that arise will be a consequence of the optimality prior.

Ultimately one needs to determine the appropriate model type on a case-by-case basis. However, only the Wang model was considered here.

5.3.4 Extending The Topology

The topology of the network - both at the population level, and at the individual neuron level - is crude. At the population level, there is only one inhibitory population. Head-direction cells implement a so-called "bump attractors" on a ring: the neurons encoding the direction the head is facing can be represented as sitting on the ring, with the direction the head is currently facing being represented as local increased activity on the ring. This can be used to implement a continuous action-selection approach [79, 80, 81], but relies on an architecture of local activation and non-uniform decreasing distal inhibition [67, 38] which cannot be implemented with one inhibitory population. Adding multiple inhibitory populations might allow one to implement this.

Moreover, the inability of the CMA-ES algorithm to find solutions to the XOR task suggests that this architecture can only solve linearly separable problems Allowing for more inhibitory populations may extend the range of solvable problems.

At the individual neuron level, the all-to-all connectivity is simply unrealistic. Common alterations to this is to connect each potential synapse with a fixed small random probability p_{conn} or by choosing a fixed number of inputs k_{conn} to each of the N postsynaptic neurons and selecting the k_{conn} inputs uniformly at random. In either case, these synapses would be selected independently of the synaptic weights so that (for a fixed postsynaptic potential V_i) the expected input is proportional to $\sum_{j=0}^{N} \langle w_{ij} \rangle p_{conn} \langle S_j(t) \rangle \approx p_{conn} N \langle w_{ij} \rangle \nu$ in the former case or $k \langle w_{ij} \rangle \nu$ in the latter case, yielding effective the same mean-field reduction [38]. These methods have the added benefits that the neurons are even more uncorrelated, as they have non-overlapping presynaptic populations. However, the former connectivity scheme lead to Erdős–Rényi networks while the latter still leads to Poisson-distributed out-degrees [111, 112] with fixed in-degree k_{conn}, neither of which is biologically accurate [113, 114, 60, 115].

Indeed, the topology of the network seems to play an important role in determining frequency of synchronous activity [130].

Instead one might consider evolving the exact network topology alongside the learning rule, using a mechanism such as is used with the NEAT algorithm (see Section 2.4.3), or with a rule which does not impose genetic drift. However, this has two problems. Firstly it may make the

learning rule formally redundant: weight agnostic neural networks are artificial neural networks which can solve problems solely due to their architecture with little dependence on the actual weights [131], and while these networks are artificial it is conceivable that a similar phenomenon may arise in spiking networks. From a biological perspective, the human genome is estimated to have around 2×10^4 genes [132], yet the total number of synapses in the adult human neocortex may be in the order of 10^{14}, supporting the idea that the actual synapses arise from some generative process dependent on environment or randomness.

Thus, to capture a more realistic network topology, one could choose a parameterised generative model for the network such as in [133], although the generative model should depend on the synapse type yet maintain biologically observed properties such as small-world and scale-free effects [134]. Alternatively, if degree distribution is all one wishes to match, given a viable degree sequence it is possible to generate a corresponding network [111]. One could thus parameterise the degree sequence alone. In either case, these parameters might be evolved alongside the learning rules to allow for a "distributionally robust" evolved learning rule solution which does not overfit the exact network. In either case, one might still be able to determine a mean-field reduction dependent on the particular degree distributions.

The plasticity rule has another limitation which allows for further generalisation: The upper bound on synaptic strengths w_{max} (and, by extension, the weight-dependent upper bound on LTP increments being proportional to $(w_{max} - w_{ij})^{\mu}$, as adapted from [16]) disallows the heavy-tails of synaptic strengths which are experimentally observed. Indeed, while synaptic weights may be unimodal, they appear to be *lognormal* in distribution [60]. This lognormal distribution when coupled with sparse synapses allows for internally generated noise and stochastic resonance which may benefit associative memory and thus may be testable from an optimality-first approach.

Building on the observations of [65] (and discussed in [16]), that LTD decrements depend linearly on the current synaptic strength while LTP increments depend only weakly on the current synaptic strength, one might propose two exponents to capture these varying dependencies: μ_{ltd} for LTD and μ_{ltp} for LTP. To match the experimental data of [65], μ_{ltd} might be set to 1. If so, [60] show that allowing LTD to be strongly weight-dependent only when the weights are small, and weakly weight-dependent for synaptic weights larger than a chosen threshold, yields a lognormal distribution of synaptic weights.

5.4 Limitations

The approximation used in Section 2.2.5 that the three-factor learning rule dynamics $H_3(M, \text{pre}, \text{post}; w(t))$ can be approximated by $g(M(t))H_2(\text{pre}, \text{post}; w(t))$ or $R(t)H_2(\text{pre}, \text{post}; w(t))$ is crude, and presents a strong limitation of the range of potential dynamics (both of the plasticity and of the network)

that can be captured by the procedure in this thesis. Nonetheless, unlike potential changes discussed in Section 5.3, the limitations arising from this, which I will discuss here, cannot so easily be overcome.

It may be true that given a vector of neuromodulator concentrations M, where individual components may be represented by dopamine, norepinephrine, acetylcholine or any modulator which may influence plasticity, the relevant projection of the neuromodulatory signal for a normative theory is the reward $R(t) = g(M(t))$, one must nevertheless keep in mind that what is investigated in such a normative approach is *reward*-driven plasticity and not neuromodulator-driven plasticity. Even so, it is clear that there are memory effects with application of dopamine: for example, applying dopamine after the plasticity induction protocol can convert LTD to LTP [135]. Moreover, distinct neuromodulators can act in concert to help guide plasticity towards solving RL tasks: an increase in cholinergic activity is correlated with exploratory behaviour, and the acetylcholine can encourage the same synapses to undergo LTD as those which later may instead undergo LTP if a dopaminergic signal arrives [136].[4]

We might therefore better think of plasticity of these synapses as a state-dependent dynamical system where the states may be "exploring", "receiving reward", and "other", where the method considered here crudely approximates this a continuum over two states characterised by a reward signal $R(t)$. On the other hand, it is difficult to construct a normative theory for the other states. Determining what the plasticity dynamics optimally should be when the agent is exploring is coupled to determining when the agent should optimally be exploring, a difficult problem related to the exploration-exploitation trade-off [82].

We know that dopaminergic activity at least correlates with RPEs. In Section 2.2.5 I observed that including M, even it is one-dimensional, in the Volterra series massively increases the dimensionality of the search space. Nonetheless, the memory effects of dopamine on plasticity requires that it is included in the Volterra series, yet to do so practically or in a computationally efficient manner using EAs may prove to be a challenge.

It may also be that there are multiple eligibility traces with different timescales, or that the individual eligibility traces are higher dimensional with different components evolving on different timescales. In [27, 28], distinct eligibility traces are considered for LTP and LTD. These distinct eligibility traces appear interact with neuromodulatory signals in distinct ways [27]. Furthermore, such a separation of eligibility traces and the consequent competition between (driving potentiation and depression, respectively) can allow for stable learning by implementing a stopping rule so that learning terminates even when the reward is present. This further supports the approach of [26]

[4]This is conceptually similar to the optimism under uncertainty principle of RL whereby the state values or state-action values are estimated more highly for unexplored states or pairs of states and actions so that once it is explored under a stochastic policy such as an ϵ-greedy search, the probability of exploring it again decreases, while visiting unexplored states is encouraged [82]; however, in this case the state to be explored is increased postsynaptic activity, and the probability of it occurring again decreases under LTD.

in implementing distinct traces.

Finally, as explored in [76] and reviewed in [73], it may be important that the reward signal has zero-mean. This amounts to replacing $R(t)$ with $R(t) - \langle R(t) \rangle$ or in the case of allowing multiple states in the environment (or solving multiple simultaneous rewarding task) $R(t) - \langle R(t)|\text{state} \rangle$ where $\langle R(t) \rangle$ or $\langle R(t)|\text{state} \rangle$ may be low-pass filtered traces of prior rewards or determined by a critic model, as in [79] and [81]. Some running average or critic needs to be implemented alongside the model to allow for these changes.

Finally, going back to the idea of state-dependent plasticity dynamics, we might ask, "Would it not be optimal if the network dynamics are state-dependent too?" Indeed, the origin of the Wang model was in [19], an investigation into how dopamine's effects on NMDA activity can bring about persistent elevated activity. Dopamine can affect NMDAR-modulated PSPs [19], or directly influence neurotransmitter release [137]. Thus neuromodulators can alter the activity at the synapse independently of plasticity. At the very least, the synaptic gating variables $s_{rec}(t)$ should be given neuromodulator dependence:

$$\frac{ds_{rec}(t)}{dt} = f_{rec}(M(t), s_{rec}(t), S_j(t))$$

for some dynamics f_{rec}.

Chapter 6

Conclusion

conticuit tandem, factoque hic fine quievit
silent at last, he ceased, and took repose

Vergil, Aenead, book III line 716
Theodore C. Williams translation,
found at Perseus

It is difficult to complete a project such as this one that feels so ripe for further investigation. Nonetheless, it must be done.

Despite decades of research, a full comprehension of synaptic plasticity eludes us. This has been exacerbated by the observations that, in increasingly complex ways, neuromodulation can change the course of plasticity, including via retroactive gating [135, 136, 3, 5]. Moreover, even one of the most studied neuromodulatory effects - dopamine release by VTA neurons - is not limited to signaling an RPE but may also signal novelty and more [138, 3]. Attempting to disentangle these plasticity phenomena one experiment at a time may prove a daunting task. Adopting a theory-first approach whereby potential roles and effects of neuromodulatory signals, and trajectories of synaptic strengths, are predicted and tested may be more expedient.

Synaptic plasticity and decision making are important areas of study particularly for understanding maladaptive behaviour and psychiatric disorders such as addiction, autism and depression [139, 140]. On the one hand, addiction and some of its behavioural consequences might arise from a hijacked reward prediction system [6, 141, 140]. On the other hand, LTP and LTD in the VTA and nucleus accumbens are implicated in addiction [142]. Altered decision making is implicated in depression, addiction and autism (this latter particularly in the context of social games) [139]. Synaptic plasticity may also explain ordinary phenomena, such as exploration and matching behaviour [138, 83, 84]. Such explanations, however, would require bridging the gap between (changes in) the biophysical inner workings of the brain and (changes in) the observed macroscopic

behaviours and choices of the organism.

Primarily I have endeavoured to provide a means to explore biologically feasible plasticity rules using the optimisation procedure CMA-ES. The learning rules considered were of the STDP type but implemented in a rate-based framework; for the task at hand, this provides little loss of generality as the evaluation procedure depended explicitly on the firing rate at a fixed time and required no sequential temporal dynamics. However, even for tasks where the spike timing would be relevant, this method may yield an initial estimate of a corresponding plasticity rule as it captures the average dynamics of such a rule. From the rate-based rule one can return to the spiking rule with exponential kernels, albeit with a certain redundancy of parameters: in a sense the decay times of the exponential determine the timespan over which the rate-based rule approximates the spiking rule, thus fixing a time constant and choosing the other parameters accordingly should allow one to arrive at a fitted STDP rule. This leaves open the question of whether such a procedure would work, or if there is some gross misstep in logic of using a reduced model in making inference about a more finely-grained model, or even if the parameters in the spiking model are in some sense unidentifiable. Unfortunately, I did not have time to explore this thread of identifiability analysis.

I have also attempted to address the issue of whether such a normative exploration provides value, and it does in two ways. First, this guides further research, helping us to filter out the infeasible from the feasible and providing guidance for further experiments. Secondly, the results can be incorporated in a Bayesian framework to provide information in their own right: we can ask questions such as, "From what distribution would a biologically accurate version of [some model] sample its parameters?" under the assumption that biology and evolution are driven by dynamics with a potential at least locally proportional to performance on the task at hand. Learning and synaptic plasticity exist for survival and mostly, if not only, for the act of improving at tasks; thus inheriting the performance measure from a reinforcement learning framework is only natural.

Finally, and possibly most importantly, I have attempted to bring together various streams of research often studied in isolation. Decision making literature discusses the decision making process with abstract decision variables and parameters which may change over time. The Wang model unites that process with a model of neural activity, although other attractor models with more (realistic) features - short-term synaptic dynamics, more realistic network topology, variability of synaptic delays and distributions of synaptic strengths, among others - would work too. The key product of this union is that we can interrogate the biophysical in light of the psychological. But decision making is inherently tied to learning, especially on repeated tasks where performance improves as in the domain of RL, and so the next step is to incorporate biologically feasible models of potentially reward-driven plasticity. Altogether, this work combines a biophysical neural network model, chosen for its compatibility with decision making experiments, with synaptic plasticity and an optimisation driven approach to determine a priori what synaptic dynamics in the presence of

reward should look like. With the framework outlined here, one can collect evidence of a decision making process and, in a maximum a posteriori sense and using a global optimisation procedure of one's own choice, determine the plasticity rule parameters which give rise to the observed learning process. Thankfully, there is much still to explore here.

Appendix A

Appendix

A.1 Description of Wang Model

In this section I will outline the Wang model of 2.3.3.

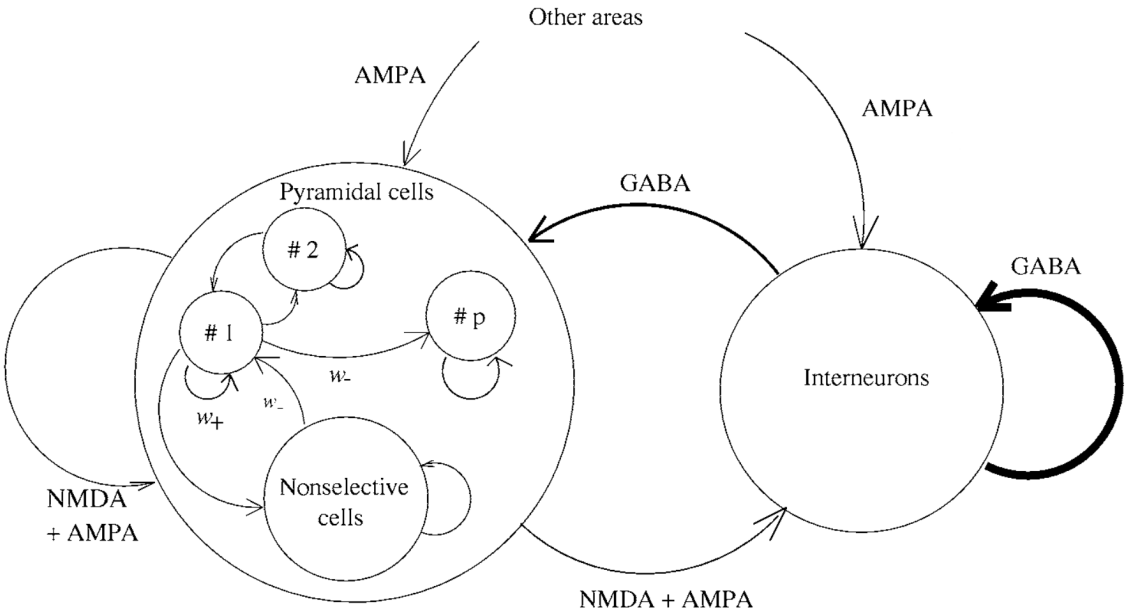


Figure A.1: The original Wang model, implemented with p selective populations. In the rate-based model, each population rate becomes a single dynamic variable. In the SNN, each population consists of many individual spiking neurons and the population rate is determined by averaging over the spike times of the neurons within the population. Image taken and adapted from [19].

A.1.1 Neurons and Synapses

The Wang model is a conductance-based LIF SNN where the fraction of open synaptic channels at a synapse are modeled as dynamic variables as discussed in 2.1.

A.1.2 Network Topology

As a point-neuron model, the synapse between any two neurons stands in place of the multitude of true (possibly discretely strengthened) synapses. Initially the synaptic strengths are fixed at a constant for each pair of populations.

The network topology is all-to-all. Nonetheless, there is also structure in the synaptic strengths. Three synaptic strengths were used in the original model, $w_+, w_-, 1$. w_+ and w_- were varied in concert so as to maintain a fixed mean total synaptic input to all the cells. These values were distributed as follows: the full excitatory population of cells were divided into p selective populations plus a remaining non-selective population, where the selective populations all consisted of a fraction f of the total number of excitatory neurons. The synapses between the cells within the same selective population had a stength of w_+, while between excitatory groups (be it selective or non-selective) the strength was w_-. Between all other neurons, the strength was 1. I adopt this same initial distribution of strengths as initial conditions, but the plasticity will not be restricted to maintain the relationships between the weights. When evolving learning rules, I start the system in a state where $w_+ = w_- = 1$.

A.1.3 External Poisson Noise

The dynamics of spiking networks can be characterised by falling into various "regimes" or classes of behaviour. One such division is the fluctuation- versus mean-driven regime [38]. The characteristic feature of the fluctuation-driven regime is that the strength of the current into/out of the cell is such that the steady-state membrane potential in (2.1) falls below the threshold V_{thr}, and hence the membrane potential only crosses the threshold when driven to by its fluctuations. In the mean-driven regime, this steady-state membrane potential falls above the threshold, repeatedly driving the neuron to cross it *despite* its fluctuations. See Figure 2.6 for an comparison. Indeed, in the absence of noise, in the mean-driven regime the neurons synchronise [68].

In the fluctuation-driven regime, the ISI distribution approaches and can be approximated by an exponential distribution. Hence this is how background noise is modelled, by Poisson processes: each neuron receives sufficient external glutamatergic Poisson input to maintain a low firing rate activity, although these external synapses do not have an NMDA component.

Synapses from these external Poisson inputs will be assumed to have constant strength of 1. Moreover, because they do not have an NMDA component, the AMPAR conductance for these inputs is scaled up to $g_{AMPA,ext}$.

The values of all these discussed parameters are given in Table A.1.

A.2 Rate Reduction of Wang Model

Here I start with the Wang Model described in A.1 and summarised in equations (2.35) to (2.41). The reduction to a rate-based model follows a mean-field approach [19, 36]. The goal is to arrive at a $(p+2)$-population rate-based model, maintaining the nonlinearities and synaptic dynamics as much as possible

It will help to have the full spiking neural network model at hand for reference:

$$\begin{aligned} C_m \frac{dV_i(t)}{dt} = &-g_m(V_i(t) - V_L) \\ &- (V_i(t) - V_E) \sum_{k=0}^{p} \Bigg[g_{AMPA} \sum_{j \in \mathcal{P}_k} w_{ij} s_{j,AMPA}(t) \\ &\qquad + g_{NMDA}(V_i(t)) \sum_{j \in \mathcal{P}_k} w_{ij} s_{j,NMDA}(t) \Bigg] \\ &- (V_i(t) - V_I) g_{GABA} \sum_{j \in \mathcal{P}_I} w_{ij} s_{j,GABA}(t) \\ &- (V_i(t) - V_E) g_{AMPA,ext} \sum_{j \in \mathcal{P}_{ext}} s_{j,AMPA}(t) \end{aligned} \tag{A.1}$$

We will focus on a neuron i in the focal population k.

The first step is to consider the synaptic weights w_{ij}. Starting with the assumption that the neurons within the same population receive statistically identical inputs, they can be treated as independent samples from independent trials. Moreover, assuming that the weights from population k' to our focal population k are identically distributed means that the dynamics of each weight can be treated as an independent sample. Moreover, assuming these dynamics are slow relative to the timescale of the firing rates, such that w_{ij} and the synaptic gating variables s_j (driven by presynaptic activity) are independent, we can in expectation decouple the products: $\langle w_{ij} s_{j,rec} \rangle = \langle w_{ij} \rangle \langle s_{j,rec} \rangle$. In brief, using

$$\langle w_{ij} \rangle_{j \in \mathcal{P}_{k'}} = \frac{1}{C_{k'}} \sum_{j \in \mathcal{P}_{k'}} w_{ij}$$

we can define

$$\langle w \rangle_{k,k'} = \frac{1}{C_k} \sum_{i \in \mathcal{P}_k} \langle w_{ij} \rangle_{j \in \mathcal{P}_{k'}} \tag{A.2}$$

In the case of the original Wang Model, these weights were fixed constants, but here we allow for variability.

Next we consider the sums of gating variables $\sum_j s_{j,rec}(t)$. We rewrite these as their mean

processes $\langle s_{rec} \rangle$ and their total deviations from the mean $\Delta S_{rec}(t)$, that is

$$\begin{aligned}
\sum_{j \in \mathcal{P}_{k'}} s_{j,AMPA}(t) &= C_{k'} \langle s_{AMPA} \rangle_{k'} + \Delta S_{AMPA,k'}(t) \\
\sum_{j \in \mathcal{P}_{ext}} s_{j,AMPA,ext}(t) &= C_{ext} \langle s_{AMPA,ext} \rangle + \Delta S_{AMPA,ext}(t) \\
\sum_{j \in \mathcal{P}_I} s_{j,GABA}(t) &= C_{k'} \langle s_{GABA} \rangle + \Delta S_{GABA}(t) \\
\sum_{j \in \mathcal{P}_{k'}} s_{j,NMDA}(t) &= C_{k'} \langle s_{NMDA} \rangle_{k'} + \Delta S_{NMDA,k'}(t)
\end{aligned} \tag{A.3}$$

The dynamics of average gating variables $\langle s_{rec} \rangle$ will be considered later. Of the noise terms, it will help to have one timescale for the synapses so that we may use the inverse mean-passage-time formula for the firing rates [36]. In the Wang Model, the dominant conductance with the shortest timescale, and thus the main source of noise, arises from the external Poisson inputs. Due to the slower timescales of the NMDA and and GABA dynamics, we can treat the variables $\Delta S_{NMDA,k'}(t)$ and $\Delta S_{GABA}(t)$ as effectively zero; Since the conductance of the external AMPA inputs is much higher, we also neglect variables $S_{AMPA,k'}(t)$ [19].

Next we approximate the deviations of the summed external gating variables $\Delta S_{AMPA,ext}(t)$ as an Gaussian process which has zero mean and correlation function

$$\langle \Delta S_{AMPA,ext}(t) \Delta S_{AMPA,ext}(t') \rangle = C_{ext} \nu_{ext} \tau_{AMPA} \exp\left(-\frac{|t - t'|}{\tau_{AMPA}} \right) \tag{A.4}$$

Putting this together, we have

$$\begin{aligned}
C_m \frac{dV_i(t)}{dt} = &-g_m(V_i(t) - V_L) \\
&- (V_i(t) - V_E) \sum_{k'=0}^{p} \langle w \rangle_{k,k'} \left[g_{AMPA} C_{k'} \langle s_{AMPA} \rangle_{k'} \right. \\
&\qquad \left. + g_{NMDA}(V_i(t)) C_{k'} \langle s_{NMDA} \rangle_{k'} \right] \\
&- (V_i(t) - V_I) \langle w \rangle_{k,I} \, g_{GABA} C_I \langle s_{GABA} \rangle \\
&- (V_i(t) - V_E) g_{AMPA,ext} C_{ext} \langle s_{AMPA,ext} \rangle \\
&- (\cancelto{\langle V \rangle}{V_i(t)} - V_E) g_{AMPA,ext} \Delta S_{AMPA,ext}(t)
\end{aligned} \tag{A.5}$$

where I have replaced the driving force of the noise term in the final line with its average, so that the full noise component can be handled neatly.

Next we turn to the nonlinear conductance and driving force for the NMDAR ion channels. We have

$$(V_i(t) - V_E) g_{NMDA}(V_i(t)) = \frac{(V_i(t) - V_E)}{1 + \gamma \exp(-\beta V_i(t))}$$

We can linearise this expression around the same average membrane potential used for the noise term $\langle V \rangle$, yielding

$$\frac{(V(t)-V_E)}{1+\gamma_{JS}\exp(-\beta_{JS}V(t)} \approx \frac{\langle V\rangle - V_E}{J(\langle V\rangle)} + (V(t)-\langle V\rangle)\underbrace{\left(\frac{1}{J(\langle V\rangle)} + \beta\frac{(\langle V\rangle - V_E)(J(\langle V\rangle)-1)}{J(\langle V\rangle)^2}\right)}_{=:J_2(\langle V\rangle)}$$

where $J(\langle V\rangle) = 1+\gamma_{JS}\exp(-\beta_{JS}\langle V\rangle)$, $J_2(\langle V\rangle)$ is the term in the rightmost parentheses, and terms at least quadratic in $(V_i(t)-\langle V\rangle)$ are dropped. The linearisation is close for a feasible range of membrane potentials (see Figure A.2).

From here we can define an effective conductance and an effective reversal potential for the NMDARs [36]:

$$g^{eff}_{NMDA}(\langle V\rangle) = g_{NMDA}J_2(\langle V\rangle) \tag{A.6}$$

$$\begin{aligned} V^{eff}_E(\langle V\rangle) &= \langle V\rangle - \frac{g_{NMDA}}{g^{eff}_{NMDA}(\langle V\rangle)}\left(\frac{\langle V\rangle - V_E}{J(\langle V\rangle)}\right) \\ &= \langle V\rangle - \frac{1}{J_2(\langle V\rangle)}\left(\frac{\langle V\rangle - V_E}{J(\langle V\rangle)}\right) \end{aligned} \tag{A.7}$$

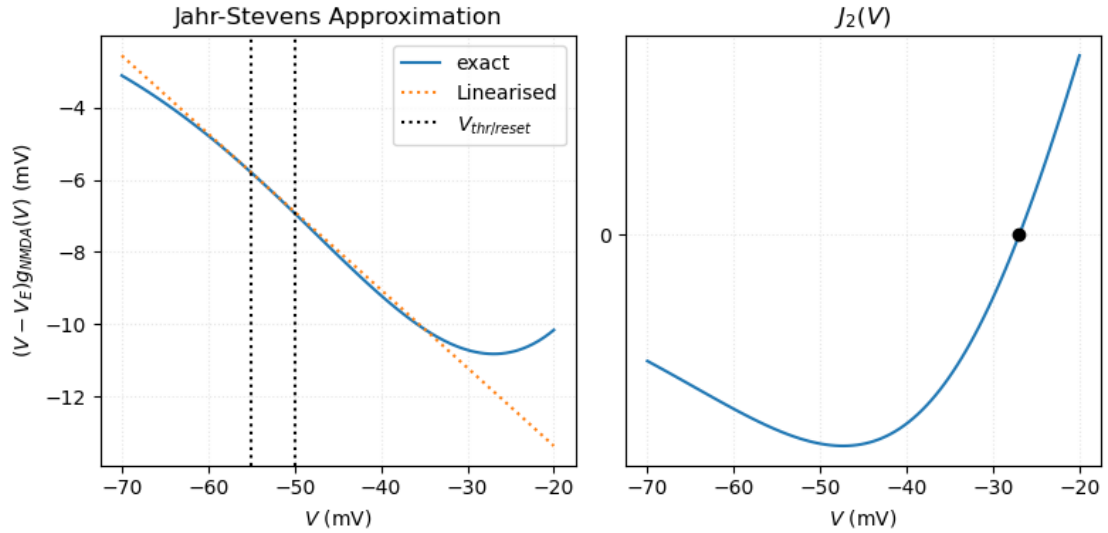


Figure A.2: Jahr-Stevens Linearisation. On the left we see the linearisation of the Jahr-Stevens formula for NMDA channel conductance, linearised around -55mV. The true membrane potential values must remain beneath $V_{thr} = -50$mV and should not deviate far beneath $V_{reset} = -55$mV, cf. Figure A.3. On the right is the shape of $J_2(V)$, where it achieves zero near -27mV. This can cause numerical errors when computing $V^{eff}_E(\langle V\rangle)$ using the formula in (A.7).

We can achieve the Gaussian process for $(\langle V\rangle - V_E)g_{AMPA,ext}\Delta S_{AMPA,ext}(t)$ with an Ornstein-Uhlenbeck process ΔI_k with dynamics

$$\tau_{AMPA}\frac{d\Delta I_k(t)}{dt} = -\Delta I_k(t) + \sigma^{eff}_C(\langle V\rangle)\eta(t), \tag{A.8}$$

where $\eta(t)$ is a Gaussian white noise process with an (effective) diffusion coefficient given by

$$\begin{aligned}\sigma_C^{eff}(\langle V\rangle) &= \sqrt{g_{AMPA,ext}^2(\langle V\rangle - V_E)^2 C_{ext}\,\langle s_{AMPA,ext}\rangle\,\tau_{AMPA}} \\ &= \sigma_V^{eff}(\langle V\rangle)\frac{C_m}{\sqrt{\tau_m^{eff}(\langle V\rangle)}},\end{aligned} \tag{A.9}$$

where σ_V^{eff} is used in the first-passage time formula (A.16) and τ_m^{eff} is defined below.

Including this in (A.5), we can write the full Langevin dynamics as

$$\begin{aligned}C_m\frac{dV_i(t)}{dt} = &-g_m(V_i(t)-V_L) \\ &-(V_i(t)-V_E)\sum_{k'=0}^{p}\langle w\rangle_{k,k'}\,g_{AMPA}C_{k'}\,\langle s_{AMPA}\rangle_{k'} \\ &-(V_i(t)-V_E^{eff}(\langle V\rangle))\sum_{k'=0}^{p}\langle w\rangle_{k,k'}\,g_{NMDA}^{eff}(\langle V\rangle)C_{k'}\,\langle s_{NMDA}\rangle_{k'} \\ &-(V_i(t)-V_I)\,\langle w\rangle_{k,I}\,g_{GABA}C_I\,\langle s_{GABA}\rangle \\ &-(V_i(t)-V_E)g_{AMPA,ext}C_{ext}\,\langle s_{AMPA,ext}\rangle \\ &-\Delta I_k(t)\end{aligned} \tag{A.10}$$

If we determine the effective time constant and effective leak conductance as

$$\tau_m^{eff} = \frac{C_m}{g_m^{eff}} = \tau_m\frac{g_m}{g_m^{eff}} \tag{A.11}$$

$$\begin{aligned}g_m^{eff} =& g_m + \sum_{k'=0}^{p}\langle w\rangle_{k,k'}\,g_{AMPA}C_{k'}\,\langle s_{AMPA}\rangle \\ &+\sum_{k'=0}^{p}\langle w\rangle_{k,k'}\,g_{NMDA}^{eff}C_{k'}\,\langle s_{NMDA}\rangle \\ &+\langle w\rangle_{k,I}\,g_{GABA}C_I\,\langle s_{GABA}\rangle \\ &+g_{AMPA,ext}C_{ext}\,\langle s_{AMPA,ext}\rangle\end{aligned} \tag{A.12}$$

then we can determine the steady-state membrane potential (that is, the membrane potential that would be achieved in the absence of neuron i spiking) as

$$\begin{aligned}V_{SS} =& \frac{g_m}{g_m^{eff}}V_L + \sum_{k'=0}^{p}\langle w\rangle_{k,k'}\,\frac{g_{AMPA}}{g_m^{eff}}C_{k'}\,\langle s_{AMPA}\rangle\,V_E \\ &+\sum_{k'=0}^{p}\langle w\rangle_{k,k'}\,\frac{g_{NMDA}^{eff}}{g_m^{eff}}C_{k'}\,\langle s_{NMDA}\rangle\,V_E^{eff} \\ &+\langle w\rangle_{k,I}\,\frac{g_{GABA}}{g_m^{eff}}C_I\,\langle s_{GABA}\rangle\,V_I \\ &+\frac{g_{AMPA,ext}}{g_m^{eff}}C_{ext}\,\langle s_{AMPA,ext}\rangle\,V_E\end{aligned} \tag{A.13}$$

and hence rewrite the membrane potential Langevin dynamics (A.10) as

$$\begin{aligned} \tau_m^{eff} \frac{dV_i(t)}{dt} &= -(V_i(t) - V_{SS}) + \frac{\Delta I_k(t)}{g_m^{eff}} \\ \tau_{AMPA} \frac{d\Delta I_k(t)}{dt} &= -\Delta I_k(t) + \sigma_C^{eff} \eta(t) \end{aligned} \tag{A.14}$$

where (A.8) is repeated for reference and dependence on $\langle V \rangle$ is suppressed.

The average membrane potential can be computed using the following formula [36]:

$$\langle V \rangle = V_{SS} - (V_{thr} - V_{reset}) \nu \tau_m^{eff}(\langle V \rangle) - (V_{SS} - V_{reset}) \nu \tau_{refrac} \tag{A.15}$$

which also depends on the effective membrane time constant $\tau_m^{eff}(\langle V \rangle)$ (see Figure A.3).

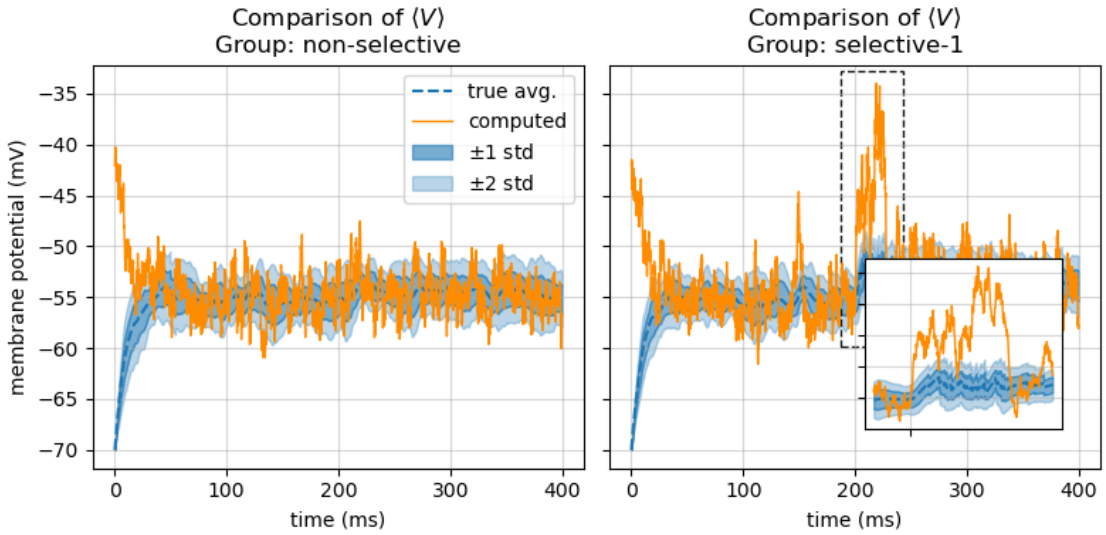


Figure A.3: Comparison of average membrane potential formula with true distribution. A spiking simulation was run with a increased Poisson input to selective population 1 at time 200ms. The analytic formula for the average membrane potential (A.15) is compared to the true average with standard deviations. Shortly after the initialisation of the simulation, the formula gives a good approximation. However, after a change in input (200-225ms, right hand plot) the formula becomes inaccurate: here the true membrane potential of any neuron cannot rise above $V_{thr} = -50$mV. Notice that the average membrane potential estimate verges on the zero value of $J_2(\langle V \rangle)$, cf. Figure A.2

Now that we have the Langevin dynamics (A.14) we can use the result built on Fokker-Planck theory which gives us the first-passage time formula [19, 36, 38] also known as the Siegert formula [143]:

$$\tau_{refrac} + \tau_m^{eff} \sqrt{\pi} \int_{(V_{reset} - V_{SS})/\sigma_V^{eff}}^{(V_{thr} - V_{SS})/\sigma_V^{eff}} \exp(x^2)(1 + \mathrm{erf}(x))dx \tag{A.16}$$

This formula gives us the expected time until next firing of a neuron, and is exact when the noise is a Gaussian white noise process, but in our case the noise has autocorrelations decaying with τ_{AMPA}. To use this formula, we need to adapt the bounds of the integral [36]. To do this I

follow [19] and change the upper bound of the integral to

$$\begin{aligned}\text{upperbound}(V_{SS}, \sigma_V^{eff}, \tau_m^{eff}) :=& \frac{V_{thr} - V_{SS}}{\sigma_V^{eff}} \left(1 + 0.5\frac{\tau_{AMPA}}{\tau_m^{eff}}\right) \\ &+ 1.03\sqrt{\frac{\tau_{AMPA}}{\tau_m^{eff}}} - 0.5\frac{\tau_{AMPA}}{\tau_m^{eff}}\end{aligned} \tag{A.17}$$

As the time of the noise decreases i.e. $\tau_{AMPA} \to 0$ we recover the original bound.

Inverting this formula gives us the rate at which neurons cross the firing threshold, or the firing rate.

A.2.1 Current-Based Approximation

The formula (A.16) depends on $\tau_m^{eff}, \sigma_V^{eff}$ as well as the steady state membrane potential V_{SS}. These all depend on the average membrane potential $\langle V \rangle$ which, through equation (A.15), depends on the firing rate. Thus this formula needs to be computed self-consistently, which is time consuming.

To avoid this, I make two approximations: I find a value V_{drive} to replace $V_i(t)$ in the driving force terms $(V_i(t) - V_{E/I})$, which yields a current-based model (see Section 3.5.1) and I replace the use of the average membrane potential in the noise strength (A.9) with the same V_{drive}, which I shall denote σ_V. The noise can still change by modulating the rate of the external inputs, but it will no longer change as a function of the population firing rates. For the conductance based model, the effective parameters (aside from (A.6) and (A.7)) fall away.

For this model we have $V_{SS} = V_L - I/g_m^{eff}$, so we can write the in terms of I and σ_V:

$$\phi(I, \sigma_V) := \left[\tau_{refrac} + \tau_m\sqrt{\pi}\int_{(V_{reset}-V_{SS})/\sigma_V}^{\text{upperbound}(V_{SS},\sigma_V,\tau_m)} \exp(x^2)(1 + \text{erf}(x))dx\right]^{-1} \tag{A.18}$$

which gives us the value to which the population firing rate will converge given a fixed input I. When the neurons are firing irregularly, and thus the true population membrane potential is spread out, the population firing rate responds rapidly to a change in stimulus [38]. I will use this fact to make the model dynamic.

A.2.2 Creating a Dynamic Model

The mean-field model from above allows us to estimate the stead-state activity, but what we seek is a dynamic model. To this end, we treat the firing rates ν_k as well as the average fraction of open

ion channels $\langle s_{rec} \rangle$ as dynamic variables.

A population of neurons firing asynchronously responds rapidly to a change in input [38]. As such, I use a small time constant τ_r and assume that the rates for each population converge linearly on their would-be steady-state rate:

$$\tau_r \frac{d\nu}{dt} = -\nu + \phi(I, \sigma_V) \tag{A.19}$$

The synaptic gating variables follow similar dynamics, but use the appropriate time constant, implementing a low-pass filter of the presynaptic firing rate:

$$\tau_{AMPA/GABA} \frac{d\left\langle s_{AMPA/GABA} \right\rangle}{dt} = -\left\langle s_{AMPA/GABA} \right\rangle + \nu \tag{A.20}$$

The NMDA-mediated dynamics however are much slower and nonlinear. Their steady-state value can be computed explicitly as a function of the presynaptic rate ν [19, 36]:

$$\psi(\nu) = \frac{\nu\tau_{NMDA}}{1 + \nu\tau_{NMDA}} \left(1 + \frac{1}{1 + \nu\tau_{NMDA}} \sum_{n=1}^{\infty} \frac{(-\alpha\tau_{NMDA,rise})^n T_n(\nu)}{(n+1)!} \right)$$

which in turn depends on the terms

$$T_n(\nu) = \sum_{m=0}^{n} (-1)^m \binom{n}{m} \frac{\tau_{NMDA,rise}(1 + \nu\tau_{NMDA})}{\tau_{NMDA,rise}(1 + \nu\tau_{NMDA}) + m\tau_{NMDA,decay}}$$

where $\tau_{NMDA} = \alpha\tau_{NMDA,rise}\tau_{NMDA,decay}$

Due to the fast rise depending on the firing rate, by comparison with (2.8) we would expect the dynamics of $\langle s_{NMDA} \rangle$ to be of the form [19]:

$$\frac{d\left\langle s_{NMDA} \right\rangle}{dt} = -\frac{\left\langle s_{NMDA} \right\rangle}{\tau_{NMDA}} + F(\nu)(1 - \left\langle s_{NMDA} \right\rangle) \tag{A.21}$$

where F captures the influence of the rise process x_{NMDA}.

Indeed at the steady state we have that $\frac{d\langle s_{NMDA} \rangle}{dt} = 0$ and $\langle s_{NMDA} \rangle = \psi(\nu)$. This allows us to solve for F as:

$$F(\nu) = \frac{\psi(\nu)}{\tau_{NMDA}(1 - \psi(\nu))}$$

By defining $\tau^{eff}_{NMDA}(\nu) = \tau_{NMDA}(1 - \psi(\nu))$ we get the dynamics

$$\tau^{eff}_{NMDA}(\nu)\frac{d\langle s_{NMDA}\rangle}{dt} = -\langle s_{NMDA}\rangle + \psi(\nu) \tag{A.22}$$

Put together, these describe the dynamic variables of our rate model without noise.

A.2.3 Adding Extra Noise

The mean-field rate model we have arrived at is deterministic, but perceptual decision making is stochastic, as are the neural dynamics. To account for this, I follow [94] in artificially re-introducing Ornstein-Uhlenbeck noise to the input current with zero mean and a strength of $\sigma_{noise} = 0.007nA$ and time constant of τ_{AMPA} i.e.

$$\tau_{AMPA}\frac{d\Delta I}{dt} = -\Delta I + \sigma_{noise}\eta$$

where η is a Gaussian white-noise process.

This serves an added benefit: reinforcement learning relies on randomness in determining action choices. This randomness allows the learning individual to explore the space of possible choices and not be stuck greedily performing the choice which initially seemed best.

A.3 Model Parameters

Parameters for the Wang model were taken mostly from [19], with the main exception of using $p = 2$. They are also included in Table A.1.

A.4 Learning Rule Parameters

For plasticity simulations, the maximum synaptic strength w_{max} was set to 3.5 while synapses from inhibitory neurons were kept fixed at 1. The reward signal decay rate τ_{reward} was set to 1ms.

The remaining parameters which were determined by the evolutionary algorithm CMA-ES can be found in Table A.2 below. For reference, θ is the low-pass filtered postsynaptic firing rate, while ν_j, ν_i are the pre- and postsynaptic firing rates, respectively, and $\langle w_{ij}\rangle = \langle w\rangle_{kk'}$ is the relative strength of the synapse from neuron j in population k' to neuron i in population k, with a maximum value of $w_{max} > 0$.

Parameter	Description	Value
N_E	total number of excitatory neurons	800
N_I	total number of inhibitory neurons	200
f	fraction of excitatory neurons in each selective population	0.1
p	number of selective populations	2
w_+	recurrent synaptic strength for selective populations (used when plasticity was absent)	2.1
w_-	interpopulation excitatory synaptic strength (used when plasticity was absent)	$1 - f(w_+ - 1)/(1 - f)$
C_{ext}	number of external Poisson neurons impinging on each LIF neuron	800
ν_{ext}	baseline firing rate of external Poisson neurons	3 Hz
V_L	leak reversal potential	-70mV
V_L	leak reversal potential	-70mV
V_{thr}	LIF firing threshold	-50mV
V_{reset}	LIF reset potential	-55mV
V_E	reversal potential for excitatory inputs	0mV
V_I	reversal potential for inhibitory inputs	-70mV
V_{drive}	approximate membrane potential used in current-based driving force	-47.5mV
$C_{m,E}$	membrane capacitance for excitatory neurons	0.5nF
$C_{m,I}$	membrane capacitance for inhibitory neurons	0.2nF
$g_{m,E}$	leak conductance for excitatory neurons	25nS
$g_{m,I}$	leak conductance for inhibitory neurons	20nS
$g_{AMPA,ext,E}$	AMPA conductance for external-to-excitatory synapses	2.08nS
$g_{AMPA,ext,I}$	AMPA conductance for external-to-inhibitory synapses	1.62nS
$g_{AMPA,E}$	AMPA conductance for excitatory-to-excitatory synapses	$0.104\frac{800}{N_E}$nS
$g_{AMPA,I}$	AMPA conductance for excitatory-to-inhibitory synapses	$0.081\frac{800}{N_E}$nS
$g_{NMDA,E}$	NMDA conductance for excitatory-to-excitatory synapses	$0.327\frac{800}{N_E}$nS
$g_{NMDA,I}$	NMDA conductance for excitatory-to-inhibitory synapses	$0.258\frac{800}{N_E}$nS
$g_{GABA,E}$	GABA conductance for inhibitory-to-excitatory synapses	$1.25\frac{200}{N_I}$nS
$g_{GABA,I}$	GABA conductance for inhibitory-to-inhibitory synapses	$0.973\frac{200}{N_I}$nS
$\tau_{refrac,E}$	refractory time for excitatory neurons	2ms
$\tau_{refrac,I}$	refractory time for inhibitory neurons	1ms
τ_{AMPA}	AMPA synapse time constant	2ms
$\tau_{NMDA,rise}$	NMDA synapse rise process time constant	2ms
$\tau_{NMDA,decay}$	NMDA synapse time constant	100ms
τ_{NMDA}	NMDA dynamics time constant for rate model	$\alpha\tau_{NMDA,rise}\tau_{NMDA,decay}$
τ_{GABA}	AMPA synapse time constant	10ms
α	NMDA rise process coefficient	0.5kHz
γ_{jS}	parameter for Jahr-Stevens formula	1/3.57
β_{jS}	parameter for Jahr-Stevens formula	$0.062(\text{mV})^{-1}$
τ_r	time constant for rate dynamics	2ms
σ_{noise}	strength of Ornstein-Uhlenbeck current noise reintroduced into the model	0.007nA

Table A.1: Parameters used in the simulations of the Wang Model.

Parameter	Description
ξ^{00}	coefficient of the weight decay term
$\overline{\xi}_0^{ab}$	coefficient of the monomial $\nu_j^a \nu_i^b$ which does not depend on θ determined as the sum of the averaged Volterra kernels which are a-th order in the presynaptic spike train, b-th order in the postsynaptic spike train, and which do not depend on θ
$\overline{\xi}_1^{ab}$	coefficient of the monomial $\nu_j^a \nu_i^b$ which is multiplied with θ^p determined as the sum of the averaged Volterra kernels which are a-th order in the presynaptic spike train, b-th order in the postsynaptic spike train, and which do depend on θ^p
$\xi_k^{ab}(\langle w_{ij} \rangle)$	weight-dependent modified version of $\overline{\xi}_k^{ab}$ for $k = 0, 1$. Equal to $\overline{\xi}_k^{ab}(w_{max} - \langle w_{ij} \rangle)^\mu$ if $\overline{\xi}_k^{ab} > 0$, otherwise $\overline{\xi}_k^{ab} \langle w_{ij} \rangle^\mu$
$p_{decay} > 0$	exponent of θ for multiplicative weight decay
$p > 1$	exponent of θ for superlinear dependence on θ required for BCM theory and multiplied with $\xi_{ab}^1(\langle w_{ij} \rangle)\nu_j^a \nu_i^b \quad \forall ab \in \{10, 01, 20, 02, 11, 21, 12\}$
$\mu \in [0, 1]$	exponent for $w_{max} - \langle w_{ij} \rangle$ or $\langle w_{ij} \rangle$ for weight-dependence in $\xi_k^{ab}(\langle w_{ij} \rangle)$
$\beta \in [0, 1]$	linear interpolation parameter for adapting the balance between reward-driven (at $\beta = 0$) and unsupervised learning (at $\beta = 1$)
$\tau_e > 0$	decay rate of the eligibility trace for the three-factor learning rule
$\tau_\theta > 0$	decay rate of the low-pass filtered postsynaptic firing rate θ

Table A.2: Learning rule parameters fitted by the evolutionary algorithm

Bibliography

[1] R. Graves, *The Greek Myths:(Penguin Classics Deluxe Edition)*. Penguin, 2012.

[2] V. Pawlak, J. R. Wickens, A. Kirkwood, and J. N. Kerr, "Timing is not everything: neuromodulation opens the stdp gate," *Frontiers in synaptic neuroscience*, vol. 2, p. 146, 2010.

[3] W. Gerstner, M. Lehmann, V. Liakoni, D. Corneil, and J. Brea, "Eligibility traces and plasticity on behavioral time scales: experimental support of neohebbian three-factor learning rules," *Frontiers in neural circuits*, vol. 12, p. 53, 2018.

[4] P. R. Roelfsema and A. Holtmaat, "Control of synaptic plasticity in deep cortical networks," *Nature Reviews Neuroscience*, vol. 19, no. 3, p. 166, 2018.

[5] Z. Brzosko, S. B. Mierau, and O. Paulsen, "Neuromodulation of spike-timing-dependent plasticity: past, present, and future," *Neuron*, vol. 103, no. 4, pp. 563–581, 2019.

[6] P. R. Montague, P. Dayan, and T. J. Sejnowski, "A framework for mesencephalic dopamine systems based on predictive hebbian learning," *Journal of neuroscience*, vol. 16, no. 5, pp. 1936–1947, 1996.

[7] W. Schultz, P. Dayan, and P. R. Montague, "A neural substrate of prediction and reward," *Science*, vol. 275, no. 5306, pp. 1593–1599, 1997.

[8] W. Schultz, "The reward signal of midbrain dopamine neurons," *Physiology*, vol. 14, no. 6, pp. 249–255, 1999.

[9] W. Schultz, "Updating dopamine reward signals," *Current opinion in neurobiology*, vol. 23, no. 2, pp. 229–238, 2013.

[10] A. Pérez-Escudero, M. Rivera-Alba, and G. G. de Polavieja, "Structure of deviations from optimality in biological systems," *Proceedings of the National Academy of Sciences*, vol. 106, no. 48, pp. 20544–20549, 2009.

[11] W. Mlynarski, M. Hledik, T. R. Sokolowski, and G. Tkacik, "Statistical analysis and optimality of biological systems," *BioRxiv*, p. 848374, 2019.

[12] J. He and X. Yu, "Conditions for the convergence of evolutionary algorithms," *Journal of systems architecture*, vol. 47, no. 7, pp. 601–612, 2001.

[13] K. A. De Jong, *Evolutionary Computation: A Unified Approach.* MIT Press, 2006.

[14] M. Mitchell, *An introduction to genetic algorithms.* MIT Press, 1998.

[15] H. Z. Shouval, M. F. Bear, and L. N. Cooper, "A unified model of nmda receptor-dependent bidirectional synaptic plasticity," *Proceedings of the National Academy of Sciences*, vol. 99, no. 16, pp. 10831–10836, 2002.

[16] A. Morrison, M. Diesmann, and W. Gerstner, "Phenomenological models of synaptic plasticity based on spike timing," *Biological cybernetics*, vol. 98, no. 6, pp. 459–478, 2008.

[17] X.-J. Wang, "Decision making in recurrent neuronal circuits," *Neuron*, vol. 60, no. 2, pp. 215–234, 2008.

[18] R. Bogacz, E. Brown, J. Moehlis, P. Holmes, and J. D. Cohen, "The physics of optimal decision making: a formal analysis of models of performance in two-alternative forced-choice tasks.," *Psychological review*, vol. 113, no. 4, p. 700, 2006.

[19] N. Brunel and X.-J. Wang, "Effects of neuromodulation in a cortical network model of object working memory dominated by recurrent inhibition," *Journal of computational neuroscience*, vol. 11, no. 1, pp. 63–85, 2001.

[20] X.-J. Wang, "Probabilistic decision making by slow reverberation in cortical circuits," *Neuron*, vol. 36, no. 5, pp. 955–968, 2002.

[21] N. Hansen, D. V. Arnold, and A. Auger, "Evolution strategies," in *Springer handbook of computational intelligence*, pp. 871–898, Springer, 2015.

[22] J.-P. Pfister and W. Gerstner, "Beyond pair-based stdp: A phenomenological rule for spike triplet and frequency effects," tech. rep., MIT Press Cambridge, 2006.

[23] J.-P. Pfister and W. Gerstner, "Triplets of spikes in a model of spike timing-dependent plasticity," *Journal of Neuroscience*, vol. 26, no. 38, pp. 9673–9682, 2006.

[24] A. Soltani and X.-J. Wang, "A biophysically based neural model of matching law behavior: melioration by stochastic synapses," *Journal of Neuroscience*, vol. 26, no. 14, pp. 3731–3744, 2006.

[25] B. Confavreux, F. Zenke, E. J. Agnes, T. Lillicrap, and T. P. Vogels, "A meta-learning approach to (re) discover plasticity rules that carve a desired function into a neural network," *bioRxiv*, 2020.

[26] R. R. Kerr, D. B. Grayden, D. A. Thomas, M. Gilson, and A. N. Burkitt, "Coexistence of reward and unsupervised learning during the operant conditioning of neural firing rates," *PloS one*, vol. 9, no. 1, p. e87123, 2014.

[27] K. He, M. Huertas, S. Z. Hong, X. Tie, J. W. Hell, H. Shouval, and A. Kirkwood, "Distinct eligibility traces for ltp and ltd in cortical synapses," *Neuron*, vol. 88, no. 3, pp. 528–538, 2015.

[28] M. A. Huertas, S. E. Schwettmann, and H. Z. Shouval, "The role of multiple neuromodulators in reinforcement learning that is based on competition between eligibility traces," *Frontiers in synaptic neuroscience*, vol. 8, p. 37, 2016.

[29] N. Pavlidis, O. Tasoulis, V. P. Plagianakos, G. Nikiforidis, and M. Vrahatis, "Spiking neural network training using evolutionary algorithms," in *Proceedings. 2005 IEEE International Joint Conference on Neural Networks, 2005.*, vol. 4, pp. 2190–2194, IEEE, 2005.

[30] G. López-Vázquez, M. Ornelas-Rodriguez, A. Espinal, J. A. Soria-Alcaraz, A. Rojas-Domínguez, H. Puga-Soberanes, J. M. Carpio, and H. Rostro-Gonzalez, "Evolutionary spiking neural networks for solving supervised classification problems," *Computational intelligence and neuroscience*, vol. 2019, 2019.

[31] S. Risi and K. O. Stanley, "A unified approach to evolving plasticity and neural geometry," in *The 2012 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8, IEEE, 2012.

[32] S. Beaulieu, L. Frati, T. Miconi, J. Lehman, K. O. Stanley, J. Clune, and N. Cheney, "Learning to continually learn," *arXiv preprint arXiv:2002.09571*, 2020.

[33] L. Metz, B. Cheung, and J. Sohl-dickstein, "Supervised learning of unsupervised learning rules,"

[34] L. Metz, N. Maheswaranathan, B. Cheung, and J. Sohl-Dickstein, "Meta-learning update rules for unsupervised representation learning," *arXiv preprint arXiv:1804.00222*, 2018.

[35] K. Gu, S. Greydanus, L. Metz, N. Maheswaranathan, and J. Sohl-Dickstein, "Meta-learning biologically plausible semi-supervised update rules," *bioRxiv*, 2019.

[36] A. Renart, N. Brunel, and X.-J. Wang, "Mean-field theory of irregularly spiking neuronal populations and working memory in recurrent cortical networks," *Computational neuroscience: A comprehensive approach*, pp. 431–490, 2004.

[37] D. O. Hebb, "The organization of behavior; a neuropsycholocigal theory," *A Wiley Book in Clinical Psychology*, vol. 62, p. 78, 1949.

[38] W. Gerstner, W. M. Kistler, R. Naud, and L. Paninski, *Neuronal dynamics: From single neurons to networks and models of cognition*. Cambridge University Press, 2014.

[39] C. E. Jahr and C. F. Stevens, "Voltage dependence of nmda-activated macroscopic conductances predicted by single-channel kinetics," *Journal of Neuroscience*, vol. 10, no. 9, pp. 3178–3182, 1990.

[40] E. R. Kandel, J. H. Schwartz, T. M. Jessell, D. of Biochemistry, M. B. T. Jessell, S. Siegelbaum, and A. Hudspeth, *Principles of neural science*, vol. 4. McGraw-hill New York, 2000.

[41] P. Dayan and L. F. Abbott, *Theoretical neuroscience: computational and mathematical modeling of neural systems*. Computational Neuroscience Series, 2001.

[42] R. Brette and W. Gerstner, "Adaptive exponential integrate-and-fire model as an effective description of neuronal activity," *Journal of neurophysiology*, vol. 94, no. 5, pp. 3637–3642, 2005.

[43] R. Naud, N. Marcille, C. Clopath, and W. Gerstner, "Firing patterns in the adaptive exponential integrate-and-fire model," *Biological cybernetics*, vol. 99, no. 4, pp. 335–347, 2008.

[44] E. M. Izhikevich, "Simple model of spiking neurons," *IEEE Transactions on neural networks*, vol. 14, no. 6, pp. 1569–1572, 2003.

[45] R. P. Costa, R. C. Froemke, P. J. Sjöström, and M. C. van Rossum, "Unified pre-and postsynaptic long-term plasticity enables reliable and flexible learning," *Elife*, vol. 4, p. e09457, 2015.

[46] R. P. Costa, B. E. Mizusaki, P. J. Sjöström, and M. C. van Rossum, "Functional consequences of pre-and postsynaptic expression of synaptic plasticity," *Philosophical Transactions of the Royal Society B: Biological Sciences*, vol. 372, no. 1715, p. 20160153, 2017.

[47] R. P. Costa, Z. Padamsey, J. A. D'Amour, N. J. Emptage, R. C. Froemke, and T. P. Vogels, "Synaptic transmission optimization predicts expression loci of long-term plasticity," *Neuron*, vol. 96, no. 1, pp. 177–189, 2017.

[48] T. P. Vogels, H. Sprekeler, F. Zenke, C. Clopath, and W. Gerstner, "Inhibitory plasticity balances excitation and inhibition in sensory pathways and memory networks," *Science*, vol. 334, no. 6062, pp. 1569–1573, 2011.

[49] T. P. Vogels, R. C. Froemke, N. Doyon, M. Gilson, J. S. Haas, R. Liu, A. Maffei, P. Miller, C. Wierenga, M. A. Woodin, *et al.*, "Inhibitory synaptic plasticity: spike timing-dependence and putative network function," *Frontiers in neural circuits*, vol. 7, p. 119, 2013.

[50] G. Hennequin, E. J. Agnes, and T. P. Vogels, "Inhibitory plasticity: balance, control, and codependence," *Annual review of neuroscience*, vol. 40, pp. 557–579, 2017.

[51] C. Clopath, L. Ziegler, E. Vasilaki, L. Büsing, and W. Gerstner, "Tag-trigger-consolidation: a model of early and late long-term-potentiation and depression," *PLoS Comput Biol*, vol. 4, no. 12, p. e1000248, 2008.

[52] C. Clopath and W. Gerstner, "Voltage and spike timing interact in stdp–a unified model," *Frontiers in synaptic neuroscience*, vol. 2, p. 25, 2010.

[53] C. Clopath, L. Büsing, E. Vasilaki, and W. Gerstner, "Connectivity reflects coding: a model of voltage-based stdp with homeostasis," *Nature neuroscience*, vol. 13, no. 3, p. 344, 2010.

[54] J. Rubin, D. D. Lee, and H. Sompolinsky, "Equilibrium properties of temporally asymmetric hebbian plasticity," *Physical review letters*, vol. 86, no. 2, p. 364, 2001.

[55] E. M. Izhikevich and N. S. Desai, "Relating stdp to bcm," *Neural computation*, vol. 15, no. 7, pp. 1511–1523, 2003.

[56] R. Kempter, W. Gerstner, and J. L. Van Hemmen, "Hebbian learning and spiking neurons," *Physical Review E*, vol. 59, no. 4, p. 4498, 1999.

[57] R. Kempter, W. Gerstner, and J. L. Van Hemmen, "Spike-based compared to rate-based hebbian learning," *Advances in neural information processing systems*, vol. 11, pp. 125–131, 1999.

[58] R. Kempter, W. Gerstner, and J. L. v. Hemmen, "Intrinsic stabilization of output rates by spike-based hebbian learning," *Neural computation*, vol. 13, no. 12, pp. 2709–2741, 2001.

[59] J. Feng, *Computational neuroscience: a comprehensive approach*. CRC press, 2003.

[60] J.-n. Teramae and T. Fukai, "Computational implications of lognormally distributed synaptic weights," *Proceedings of the IEEE*, vol. 102, no. 4, pp. 500–512, 2014.

[61] E. L. Bienenstock, L. N. Cooper, and P. W. Munro, "Theory for the development of neuron selectivity: orientation specificity and binocular interaction in visual cortex," *Journal of Neuroscience*, vol. 2, no. 1, pp. 32–48, 1982.

[62] L. N. Cooper, N. Intrator, B. S. Blais, and H. Z. Shouval, *Theory of cortical plasticity*. World Scientific, 2004.

[63] J. Gjorgjieva, C. Clopath, J. Audet, and J.-P. Pfister, "A triplet spike-timing–dependent plasticity model generalizes the bienenstock–cooper–munro rule to higher-order spatiotemporal correlations," *Proceedings of the National Academy of Sciences*, vol. 108, no. 48, pp. 19383–19388, 2011.

[64] M. Gilson, A. Burkitt, and L. J. Van Hemmen, "Stdp in recurrent neuronal networks," *Frontiers in computational neuroscience*, vol. 4, p. 23, 2010.

[65] G.-q. Bi and M.-m. Poo, "Synaptic modifications in cultured hippocampal neurons: dependence on spike timing, synaptic strength, and postsynaptic cell type," *Journal of neuroscience*, vol. 18, no. 24, pp. 10464–10472, 1998.

[66] W. M. Kistler and J. L. v. Hemmen, "Modeling synaptic plasticity in conjunction with the timing of pre-and postsynaptic action potentials," *Neural Computation*, vol. 12, no. 2, pp. 385–405, 2000.

[67] T. P. Vogels, K. Rajan, and L. F. Abbott, "Neural network dynamics," *Annu. Rev. Neurosci.*, vol. 28, pp. 357–376, 2005.

[68] R. E. Mirollo and S. H. Strogatz, "Synchronization of pulse-coupled biological oscillators," *SIAM Journal on Applied Mathematics*, vol. 50, no. 6, pp. 1645–1662, 1990.

[69] M. Chistiakova, N. M. Bannon, J.-Y. Chen, M. Bazhenov, and M. Volgushev, "Homeostatic role of heterosynaptic plasticity: models and experiments," *Frontiers in computational neuroscience*, vol. 9, p. 89, 2015.

[70] F. Zenke and W. Gerstner, "Hebbian plasticity requires compensatory processes on multiple timescales," *Philosophical Transactions of the Royal Society B: Biological Sciences*, vol. 372, no. 1715, p. 20160259, 2017.

[71] K. D. Miller and D. J. MacKay, "The role of constraints in hebbian learning," *Neural computation*, vol. 6, no. 1, pp. 100–126, 1994.

[72] T. D. Sanger, "Optimal unsupervised learning in a single-layer linear feedforward neural network," *Neural networks*, vol. 2, no. 6, pp. 459–473, 1989.

[73] N. Frémaux and W. Gerstner, "Neuromodulated spike-timing-dependent plasticity, and theory of three-factor learning rules," *Frontiers in neural circuits*, vol. 9, p. 85, 2016.

[74] M. A. Farries and A. L. Fairhall, "Reinforcement learning with modulated spike timing–dependent synaptic plasticity," *Journal of neurophysiology*, vol. 98, no. 6, pp. 3648–3665, 2007.

[75] E. M. Izhikevich, "Solving the distal reward problem through linkage of stdp and dopamine signaling," *Cerebral cortex*, vol. 17, no. 10, pp. 2443–2452, 2007.

[76] R. Legenstein, D. Pecevski, and W. Maass, "A learning theory for reward-modulated spike-timing-dependent plasticity with application to biofeedback," *PLoS Comput Biol*, vol. 4, no. 10, p. e1000180, 2008.

[77] D. Baras and R. Meir, "Reinforcement learning, spike-time-dependent plasticity, and the bcm rule," *Neural Computation*, vol. 19, no. 8, pp. 2245–2279, 2007.

[78] R. V. Florian, "Reinforcement learning through modulation of spike-timing-dependent synaptic plasticity," *Neural computation*, vol. 19, no. 6, pp. 1468–1502, 2007.

[79] E. Vasilaki, N. Frémaux, R. Urbanczik, W. Senn, and W. Gerstner, "Spike-based reinforcement learning in continuous state and action space: when policy gradient methods fail," *PLoS Comput Biol*, vol. 5, no. 12, p. e1000586, 2009.

[80] N. Frémaux, H. Sprekeler, and W. Gerstner, "Functional requirements for reward-modulated spike-timing-dependent plasticity," *Journal of Neuroscience*, vol. 30, no. 40, pp. 13326–13337, 2010.

[81] N. Frémaux, H. Sprekeler, and W. Gerstner, "Reinforcement learning using a continuous time actor-critic framework with spiking neurons," *PLoS Comput Biol*, vol. 9, no. 4, p. e1003024, 2013.

[82] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press, 2018.

[83] Y. Loewenstein and H. S. Seung, "Operant matching is a generic outcome of synaptic plasticity based on the covariance between reward and neural activity," *Proceedings of the National Academy of Sciences*, vol. 103, no. 41, pp. 15224–15229, 2006.

[84] Y. Loewenstein, "Robustness of learning that is based on covariance-driven synaptic plasticity," *PLoS Comput Biol*, vol. 4, no. 3, p. e1000007, 2008.

[85] K. Iigaya, Y. Ahmadian, L. P. Sugrue, G. S. Corrado, Y. Loewenstein, W. T. Newsome, and S. Fusi, "Deviation from the matching law reflects an optimal strategy involving learning over multiple timescales," *Nature communications*, vol. 10, no. 1, pp. 1–14, 2019.

[86] P. L. Bartlett and J. Baxter, "A biologically plausible and locally optimal learning algorithm for spiking neurons," *Rapport technique, Australian National University*, 2000.

[87] J. Baxter and P. L. Bartlett, "Infinite-horizon policy-gradient estimation," *Journal of Artificial Intelligence Research*, vol. 15, pp. 319–350, 2001.

[88] A. Roxin and A. Ledberg, "Neurobiological models of two-choice decision making can be reduced to a one-dimensional nonlinear diffusion equation," *PLoS Comput Biol*, vol. 4, no. 3, p. e1000046, 2008.

[89] S. Tajima, J. Drugowitsch, N. Patel, and A. Pouget, "Optimal policy for multi-alternative decisions," *Nature neuroscience*, vol. 22, no. 9, pp. 1503–1511, 2019.

[90] P. L. Smith, "Diffusion theory of decision making in continuous report.," *Psychological Review*, vol. 123, no. 4, p. 425, 2016.

[91] R. Bogacz, "Optimal decision-making theories: linking neurobiology with behaviour," *Trends in cognitive sciences*, vol. 11, no. 3, pp. 118–125, 2007.

[92] R. Bogacz and T. Larsen, "Integration of reinforcement learning and optimal decision-making theories of the basal ganglia," *Neural computation*, vol. 23, no. 4, pp. 817–851, 2011.

[93] H. Wei, Y. Bu, and D. Dai, "A decision-making model based on a spiking neural circuit and synaptic plasticity," *Cognitive neurodynamics*, vol. 11, no. 5, pp. 415–431, 2017.

[94] K.-F. Wong and X.-J. Wang, "A recurrent network mechanism of time integration in perceptual decisions," *Journal of Neuroscience*, vol. 26, no. 4, pp. 1314–1328, 2006.

[95] C.-C. Lo and X.-J. Wang, "Cortico–basal ganglia circuit mechanism for a decision threshold in reaction time tasks," *Nature neuroscience*, vol. 9, no. 7, pp. 956–963, 2006.

[96] T. Neiman and Y. Loewenstein, "Covariance-based synaptic plasticity in an attractor network model accounts for fast adaptation in free operant learning," *Journal of Neuroscience*, vol. 33, no. 4, pp. 1521–1534, 2013.

[97] D. B. Fogel, *Evolutionary computation: toward a new philosophy of machine intelligence*, vol. 1. John Wiley & Sons, 2006.

[98] K. Price, R. M. Storn, and J. A. Lampinen, *Differential evolution: a practical approach to global optimization*. Springer Science & Business Media, 2006.

[99] H.-G. Beyer and H.-P. Schwefel, "Evolution strategies–a comprehensive introduction," *Natural computing*, vol. 1, no. 1, pp. 3–52, 2002.

[100] N. Hansen, "The cma evolution strategy: A tutorial," *arXiv preprint arXiv:1604.00772*, 2016.

[101] P. Hämäläinen, A. Babadi, X. Ma, and J. Lehtinen, "Ppo-cma: Proximal policy optimization with covariance matrix adaptation," in *2020 IEEE 30th International Workshop on Machine Learning for Signal Processing (MLSP)*, pp. 1–6, IEEE, 2020.

[102] C. Igel, "Neuroevolution for reinforcement learning using evolution strategies," in *The 2003 Congress on Evolutionary Computation, 2003. CEC'03.*, vol. 4, pp. 2588–2595, IEEE, 2003.

[103] T. Salimans, J. Ho, X. Chen, S. Sidor, and I. Sutskever, "Evolution strategies as a scalable alternative to reinforcement learning," *arXiv preprint arXiv:1703.03864*, 2017.

[104] D. Wierstra, T. Schaul, T. Glasmachers, Y. Sun, J. Peters, and J. Schmidhuber, "Natural evolution strategies," *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 949–980, 2014.

[105] D. Floreano, P. Dürr, and C. Mattiussi, "Neuroevolution: from architectures to learning," *Evolutionary intelligence*, vol. 1, no. 1, pp. 47–62, 2008.

[106] K. O. Stanley and R. Miikkulainen, "Evolving neural networks through augmenting topologies," *Evolutionary computation*, vol. 10, no. 2, pp. 99–127, 2002.

[107] Y. Kassahun and G. Sommer, "Efficient reinforcement learning through evolutionary acquisition of neural topologies.," in *ESANN*, pp. 259–266, 2005.

[108] K. O. Stanley, D. B. D'Ambrosio, and J. Gauci, "A hypercube-based encoding for evolving large-scale neural networks," *Artificial life*, vol. 15, no. 2, pp. 185–212, 2009.

[109] S. Risi and K. O. Stanley, "Indirectly encoding neural plasticity as a pattern of local rules," in *International Conference on Simulation of Adaptive Behavior*, pp. 533–543, Springer, 2010.

[110] A. Soltoggio, K. O. Stanley, and S. Risi, "Born to learn: the inspiration, progress, and future of evolved plastic artificial neural networks," *Neural Networks*, vol. 108, pp. 48–67, 2018.

[111] S. N. Dorogovtsev and J. F. Mendes, *Evolution of networks: From biological nets to the Internet and WWW*. OUP Oxford, 2013.

[112] M. Newman, *Networks*. Oxford university press, 2018.

[113] D. S. Bassett and E. Bullmore, "Small-world brain networks," *The neuroscientist*, vol. 12, no. 6, pp. 512–523, 2006.

[114] E. Bullmore and O. Sporns, "Complex brain networks: graph theoretical analysis of structural and functional systems," *Nature reviews neuroscience*, vol. 10, no. 3, pp. 186–198, 2009.

[115] C. Curto and K. Morrison, "Relating network connectivity to dynamics: opportunities and challenges for theoretical neuroscience," *Current opinion in neurobiology*, vol. 58, pp. 11–20, 2019.

[116] H. Risken, *The Fokker-Planck Equation: Methods of Solution and Applications*. Springer, 1989.

[117] L. V. Kibiuk, D. Stuart, M. Miller, *et al.*, *Brain facts: A primer on the brain and nervous system*. The Society For Neuroscience, 2008.

[118] L. F. Abbott and F. S. Chance, "Drivers and modulators from push-pull and balanced synaptic input," *Progress in brain research*, vol. 149, pp. 147–155, 2005.

[119] M. Stimberg, R. Brette, and D. F. Goodman, "Brian 2, an intuitive and efficient neural simulator," *Elife*, vol. 8, 2019.

[120] F.-A. Fortin, F.-M. D. Rainville, M.-A. Gardner, M. Parizeau, and C. Gagné, "Deap: Evolutionary algorithms made easy," *Journal of Machine Learning Research*, vol. 13, no. Jul, pp. 2171–2175, 2012.

[121] S. Cavallari, S. Panzeri, and A. Mazzoni, "Comparison of the dynamics of neural interactions between current-based and conductance-based integrate-and-fire recurrent networks," *Frontiers in neural circuits*, vol. 8, p. 12, 2014.

[122] T. Schwalger, M. Deger, and W. Gerstner, "Towards a theory of cortical columns: From spiking neurons to interacting neural populations of finite size," *PLoS computational biology*, vol. 13, no. 4, p. e1005507, 2017.

[123] M. Di Volo, A. Romagnoni, C. Capone, and A. Destexhe, "Biologically realistic mean-field models of conductance-based networks of spiking neurons with adaptation," *Neural computation*, vol. 31, no. 4, pp. 653–680, 2019.

[124] C. M. Bishop, *Pattern recognition and machine learning*. springer, 2006.

[125] B. O. Fatimah, W. A. Senapon, A. M. Adebowale, *et al.*, "Solving ordinary differential equations with evolutionary algorithms," *Open Journal of Optimization*, vol. 4, no. 03, p. 69, 2015.

[126] R. E. Schapire, "The strength of weak learnability," *Machine learning*, vol. 5, no. 2, pp. 197–227, 1990.

[127] J. C. Magee and C. Grienberger, "Synaptic plasticity forms and functions," *Annual review of neuroscience*, vol. 43, pp. 95–117, 2020.

[128] R. Jolivet, F. Schürmann, T. K. Berger, R. Naud, W. Gerstner, and A. Roth, "The quantitative single-neuron modeling competition," *Biological cybernetics*, vol. 99, no. 4, pp. 417–426, 2008.

[129] G. Deco, V. K. Jirsa, P. A. Robinson, M. Breakspear, and K. Friston, "The dynamic brain: from spiking neurons to neural masses and cortical fields," *PLoS Comput Biol*, vol. 4, no. 8, p. e1000092, 2008.

[130] G. Grinstein and R. Linsker, "Synchronous neural activity in scale-free network models versus random network models," *Proceedings of the National Academy of Sciences*, vol. 102, no. 28, pp. 9948–9953, 2005.

[131] A. Gaier and D. Ha, "Weight agnostic neural networks," *arXiv preprint arXiv:1906.04358*, 2019.

[132] C. Willyard, "New human gene tally reignites debate," *Nature*, vol. 558, no. 7710, pp. 354–356, 2018.

[133] H. F. Song, H. Kennedy, and X.-J. Wang, "Spatial embedding of structural similarity in the cerebral cortex," *Proceedings of the National Academy of Sciences*, vol. 111, no. 46, pp. 16580–16585, 2014.

[134] V. P. Pastore, P. Massobrio, A. Godjoski, and S. Martinoia, "Identification of excitatory-inhibitory links and network topology in large-scale neuronal assemblies from multi-electrode recordings," *PLoS computational biology*, vol. 14, no. 8, p. e1006381, 2018.

[135] Z. Brzosko, W. Schultz, and O. Paulsen, "Retroactive modulation of spike timing-dependent plasticity by dopamine," *Elife*, vol. 4, p. e09685, 2015.

[136] Z. Brzosko, S. Zannone, W. Schultz, C. Clopath, and O. Paulsen, "Sequential neuromodulation of hebbian plasticity offers mechanism for effective reward-based navigation," *Elife*, vol. 6, p. e27756, 2017.

[137] N. X. Tritsch and B. L. Sabatini, "Dopaminergic modulation of synaptic transmission in cortex and striatum," *Neuron*, vol. 76, no. 1, pp. 33–50, 2012.

[138] S. Kakade and P. Dayan, "Dopamine: generalization and bonuses," *Neural Networks*, vol. 15, no. 4-6, pp. 549–559, 2002.

[139] P. R. Montague, R. J. Dolan, K. J. Friston, and P. Dayan, "Computational psychiatry," *Trends in cognitive sciences*, vol. 16, no. 1, pp. 72–80, 2012.

[140] X.-J. Wang and J. H. Krystal, "Computational psychiatry," *Neuron*, vol. 84, no. 3, pp. 638–654, 2014.

[141] Y. Takahashi, G. Schoenbaum, and Y. Niv, "Silencing the critics: understanding the effects of cocaine sensitization on dorsolateral and ventral striatum in the context of an actor/critic model," *Frontiers in neuroscience*, vol. 2, p. 14, 2008.

[142] J. A. Kauer and R. C. Malenka, "Synaptic plasticity and addiction," *Nature reviews neuroscience*, vol. 8, no. 11, pp. 844–858, 2007.

[143] A. J. Siegert, "On the first passage time probability problem," *Physical Review*, vol. 81, no. 4, p. 617, 1951.