Growing Synfire Chains with Triphasic Spike-Time-Dependent Plasticity

by

Amelia Waddington

Submitted in accordance with the requirements for the degree of Doctor of Philosophy.

UNIVERSITY OF LEEDS

The University of Leeds
School of Computing

May 2011

Declarations

The candidate confirms that the work submitted is her own, except where work which has formed part of jointly-authored publications has been included. The contribution of the candidate and the other authors to this work has been explicitly indicated below. The candidate confirms that appropriate credit has been given within the thesis where reference has been made to the work of others.

Some parts of the work presented in this thesis have been published in the following abstracts/articles:

Waddington, A. and Cohen, N., "Capacity of networks to develop multiple attractors through STDP", In *BMC Neuroscience 2009*, 10 (Suppl 1) (2009) P195.
This work is contained in Chapter 2 and was conceived, undertaken and documented by Amelia Waddington with Guidance from Netta Cohen

Waddington, A. Appleby, P.A. deKamps M. and Cohen, N., " Emergence of Synfire Chains with Triphasic Spike-Time-Dependent Plasticity", In *BMC Neuroscience 2011*, [Accepted] (2011)
This work is contained in Chapter 3 and was conceived jointly by all authors, the work was undertaken and documented by Amelia Waddington with Guidance from Netta Cohen, Marc deKamps and Peter Appleby

Waddington, A. Appleby, P.A. deKamps M. and Cohen, N., " Triphasic Spike-Timing-Dependent Plasticity Organizes Networks to Produce Robust Sequences of Neural Activity", *Submitted*, (2010)
This work is contained in Chapter 3 and 5 and was conceived jointly by all authors, the work was undertaken and documented by Amelia Waddington with Guidance from Netta Cohen, Marc deKamps and Peter Appleby

This copy has been supplied on the understanding that it is copyright material and that no quotation from the thesis may be published without proper acknowledgement.

©2011 The University of Leeds and Amelia Waddington

Acknowledgements

Thanks to my supervisors, Netta Cohen and Marc de Kamps, who have taught me a great deal and provided constant support. Thanks to Peter Appleby who suggested triphasic STDP as a topic for research. Thanks to Seth Bullock for the encouragement at critical points. Personal thanks to John Greenall, without your support, it would not have been possible.

Abstract

How collections of neurons combine into functional networks capable of intricate and accurate information processing is one of the biggest and most interesting challenges in neuroscience today. To approach this challenge, it is necessary to address the problem one structure at a time. In this thesis the focus is the development of synfire chains. Synfire chains are feed-forward neural structures which have long been suggested as a possible mechanism by which precisely timed sequences of neural activity could be generated. Precise spatiotemporal firing patterns are known to occur in the brains of many animals including, rats, mice, song birds, monkeys and humans. Such firing patterns have been linked with a wide range of behaviours including motor responses and sensory encoding. There have been many previous computational studies which address the development of synfire chains. However, they have all required either initial sparse connectivity or strong topological constraints in addition to any synaptic learning rules. Here, it is shown that this necessity can be removed. In this model, development is guided by an experimentally reported spike-timing-dependent plasticity (STDP) rule, triphasic STDP, plus activity-dependent excitability. This STDP rule, which has not been previously used in computational studies, is shown to successfully develop a synfire chain in a network of binary neurons. The width and length of the final chain can be controlled through model parameters. In addition, it is possible to embed multiple chains within one neural network. Next, the effect of triphasic STDP is investigated in a network of more realistic leaky integrate and fire neurons. Here, synfire chain development is shown to be robust in the presence of heterogeneous delays. Finally, the development is described as a random walk, creating a concrete relationship between the model parameters and final network structure.

Contents

List of Figures

List of Tables

Introduction

The brain is amazing, capable of so much, yet still very poorly understood. Understanding how brain cells (neurons) create functioning circuits which are capable of controlling movement, learning songs, even creating love and hate, might seem like a daunting task. But by exploring the neural circuits that are closest to the external observable environment, those that receive sensory inputs and those that produce motor outputs, neuroscience is beginning to answer some of these big questions.

Primary sensory circuits are the first layer of processing in the brain, converting the inputs to our senses: light, sound, touch, tastes and smells into neural signals which are then relayed into deeper areas for further processing. At this interface between brain and environment, it is possible to control the input experimentally and simultaneously observe the activity in the related neural areas. Repeated activation of particular cells in response to a particular stimulus links the stimulus to a specific neural pattern. The hope in neuroscience research is that the combination of many such links will build into a translation of the neural language.

Perhaps the most famous example of such an endeavour was undertaken by Hubel and Weisel in the fifties and sixties [59, 60]. Here, micro-electrodes were inserted into the primary visual cortex of cats so that the neural activity could be recorded. The visual input consisted of a screen with areas of light and dark. By mapping the response of different cells to different light stimuli, they could categorise cells by receptive field and specificity to stimulus orientation. In doing so they built up a comprehensive picture of how visual inputs are initially processed by the brain, a picture which has inspired many computer vision models [15, 123, 138].

The other interface between neural processing and the environment is at the output

stage. Like sensory inputs, speech and movement can be directly observed and neural correlates can be found for these observations. Much work has concentrated on understanding the neural instructions delivered by the motor cortex [79, 127, 132], as understanding this system has huge implications for neurally driven prosthetics [24, 115, 135, 136] and could therefore be of significant medical benefit.

How exactly to decode the neural signal is a matter of much debate [46, 161]. Most agree that action potentials generated by neurons (often referred to as spikes) are the building blocks of neural information transmission. Early studies such as those by Hubel and Weisel have reported an increase in the firing rate of neurons that are engaged in behaviour (e.g., [131, 153]). Such reports suggest individual spikes are not important; that information is carried in the firing rate. This is referred to as the rate code theory. Subsequent work (e.g., [62, 124]) has found that the precise time of individual spikes is important; that information is carried in the time between spikes. This is known as the temporal coding theory. In practice, it is most likely that the brain uses both these coding methods and many more beside [63, 107, 122].

Uncovering the neural circuitry responsible for observed activity patterns is challenging for a number of reasons. Firstly, scale: there are tens of thousands of neurons per mm^3 of cortex [94] each with thousands of synaptic connections [155]. Secondly, methodologies for inferring connectivity are limited to small numbers of cells. Electrophysiology and high resolution imaging studies can determine connectivity between isolated pairs of neurons [22, 134]. This is generally undertaken between neurons from different cortical areas or different layers within the same area. Mapping the entire connectivity in a local area is infeasible with currently available experimental methods. Theoretical and computational neuroscientists have been able to propose and test possible neural circuits by creating model neural networks. Notable proposals include recurrent models of working memory [17, 18], and feed-forward structures that can propagate precise timing [2, 3]. Such models allow the investigator to explore how the experimental findings from a single or small number of cells combine in networks. Ideally, such models also direct future experimental research.

Clearly the structure of neural networks will be a major factor in determining their activity pattern. But the reverse is also true, neural activity determines the development of neural structures [11, 21, 31, 44, 106, 157]. Structural change driven by neural activity is called activity-dependent plasticity. Activity-dependent plasticity is an umbrella term for many mechanisms including; the growth or retardation of dendrites or axons, changes in synaptic weight, spine[1] turnover and changes in the excitability of neurons. Under-

[1] synapses usually occur on dendritic protrusions called spines.

standing how activity drives the development of neural structures is another area where computational models can assist. In a model it is possible to isolate and study the developmental mechanism of interest, something which can't be done in biological preparations. Note that development here and throughout this thesis refers to changes in neural structure at any stage, and could be the result of learning in adults or equally the development of immature networks.

In this thesis, computational and analytical modelling are used to investigate the development of synfire chains; feed-forward networks which have been proposed [2, 3] as a structure that could generate precisely timed spike patterns. In the first chapter, the biological motivation for this research is set out and previous computational work in this area is described. In the second chapter, a model adapted from a recently proposed model [77] is developed and tested. This model is then reduced element by element, uncovering the relative roles of the constituent parts. In chapter three, an alternative model is described which uses a hitherto untested plasticity rule. The sensitivity to a range of model parameters is explored in this heavily reduced model. Chapter four describes how the developmental mechanism proposed in chapter three can also be implemented in a more biologically realistic neural model and details experimentation with this model. Finally, synfire chain development, which had been previously modelled computationally, is described analytically, illuminating the relationship between model parameters and the final network structure.

Chapter 1

Background

This chapter begins by outlining the motivation for this research, giving specific examples of neural activity patterns that have been experimentally observed. This is followed by the description of a neural structure which could be responsible for such patterns. Finally, the development of this type of structure is addressed, there previous computational work addressing such development is described.

1.1 Precisely Timed Activity Patterns

Cracking the neural code, understanding how spikes translate into behaviour, is one of the biggest challenges in neuroscience. One theory, temporal coding suggests that the precise time of spikes is very important. The discovery of repeating spatiotemporal patterns within neural recordings supports this argument. The matching of particular patterns to observed behaviour demonstrates that this is one of (possibly many) coding techniques used by the brain. Precisely timed sequences of neural activity have been reported in many different species and in many areas of the brain. In many cases, the patterns were indeed matched to observed behaviour.

Early neural recordings from monkeys found spatiotemporal patterns that repeated at above chance levels in left frontal areas when the monkeys responded to a sound and light localisation task [4]. A later study involving a localisation task [124] again found precise firing sequences here in premotor and prefontal areas. The firing sequences were

correlated with behaviour. Interestingly, sequences involving the same neurons but with different timings were associated with different behaviours indicating that the timing of spikes is a key part of the neural code in these areas. Also in the monkey motor cortex, precisely timed neural sequences were found that could be related to the velocity and direction of drawing-like hand movements [140, 141]. More recently, repeating spatiotemporal patterns have been reported in early visual areas of monkeys [8], where it was possible to determine whether the images of monkeys' faces presented were identifiable or scrambled from the specific neural pattern observed.

In humans, a magnetoencephalography (MEG)[1] study reported spatiotemporal patterns on a cortical scale that represented speech processing as distinct from general noise processing [125]. A characteristic delay was found between activation of the superiortemporal cortex and inferiorfrontal cortex when processing words and pseudowords but not non-word noise that had been matched for acoustic and spectrotemporal characteristics. Spatiotemporal patters at the cortical scale involve very many more neurons than patterns reported in animal models which typically consist of only a few spikes. However, it may be possible that such super-patterns are hierarchically constructed from many smaller patterns which can be detected at the microcircuit level.

Evidence for such hierarchical constructs has been reported in the early visual areas of cats [70]. Here, the authors looked for temporal patterns in post synaptic events distributed over the dendrites of single neurons in order to identify patterns of activity in the collective input. Such patterns were successfully identified *in vitro* (mouse primary visual cortex slice) and *in vivo* (anaesthetised cat primary visual cortex). *In vivo* sequences of patterns were often repeated in the same order, suggesting that such a hierarchical coding system may exist. In this study there was no behavioural input with which the patterns could be linked. However, an earlier study in the auditory thalamus of cats found spatiotemporal patterns both without auditory input and when it was present. Additionally, repeating patterns were found at normal temperature and when the area had been cooled [162].

In rodents, precisely timed sequences have been reported in a variety of brain areas. Spike triplets were recorded at above chance levels in rat olfactory bulb, but no relationship could be determined between the stimulus and the spike patterns [92]. In rats trained to respond to an auditory discrimination task, repeating sequences were found in auditory cortex. Just over half of these could be accurately mapped to the rats response to the task [163]. In neocortex of awake and anaesthetised rats, repeating sequences were detected within the spontaneous activity [99]. Calcium imaging can also be used to detect

[1]MEG detects local neural activity by recording the low level magnetic fields produced by dendritic activity.

spiking patterns *in vitro* [103]. Here precise temporal patterns were observed in slices of mouse primary visual cortex. The frequency of patterns was significantly reduced when glutamate receptors were blocked, suggesting that the patterns are generated by excitatory synaptic activity.

Although spatiotemporal activity patterns have been found in a range of mammalian preparations, by far the most concrete example of precisely timed neural sequences with a direct link to a specific behaviour can be found in song birds. The zebra finch is a particularly interesting song bird as they, like humans, learn stereotyped vocal behaviour usually from their parents. Zebra finch songs are constructed from unique notes separated by brief periods of silence, one or more notes can be grouped into a syllable, a specific sequence of syllables is known as a motif, motifs are repeated or strung together to make a song [111]. In adult birds, song production is primarily undertaken by the feed-forward song motor pathway; HVC (proper name) neurons control the firing of neurons in the Robust Nucleus of the Arcopallium (RA), which control the vocal output through the telencephalon [111, 173]. Sequential and precisely timed activity patterns are found in both HVC and RA and are exactly correlated to song vocalisation on a sub-millisecond timescale [53]. The neurons from HVC to RA fire one brief burst of action potentials at precisely the same time for any particular motif [53]. Conversely, RA neurons burst at many precise times during a single motif [90], but are uniquely associated with specific notes [25, 170].

There is significant evidence which suggests HVC orchestrates song production. Firstly, the learning of songs, through imitation of an adult teacher [16], is dependent on the development of the HVC to RA connections [110]. Secondly, cooling HVC reduces the speed of the song over every timescale from sub-syllabic to entire song, whilst cooling of RA has no effect [97]. This indicates that the intrinsic neural dynamics and circuitry of HVC determine the timing of a bird's song. The authors suggest that this may be by way of multiple chain like structures within HVC . Recent electrophysiological recordings from HVC show large, rapid depolarisation before burst onset, consistent with the existence of chain like structures [98]. Computational models based on collections of chains and feedback inhibition have successfully reproduced HVC behaviour in normal [76] and deafened birds [55].

It should also be noted that there have been those who refute the precise timing hypothesis, and specifically suggest that the spatiotemporal patterns reported in some of the above studies are not an indication of underlying connectivity but are a consequence of errors in the statistical analysis. It is suggested that the number of repetitions that would occur by chance are underestimated and hence the statistical significance conclu-

sion is incorrect. Estimating the expected number of pattern repetitions using the observed spike count distribution instead of assuming a Poisson distribution [121] or generating surrogate data from the original data [9, 108] found experimental observations occurred at chance levels only. However, reanalysis of original data [71], subsequent work with more advanced recording techniques [8], and further advances in statistical testing methods [51, 133, 146] suggest that precisely timed firing sequences are a genuine phenomena. With this in mind, attention now turns to how such neural activity patterns might be created.

1.2 Synfire Chains

The idea that precisely timed neural sequences could be generated by chain like structures was first proposed by Abeles in the 1980's [2, 3]. Such structures, coined synfire chains, are defined as feed-forward structures composed of multiple layers in which the activity flows from the input, sequentially through the layers, with each repetition of the input producing the same precise firing pattern (Figure 1.1). Each layer fires synchronously and the time delay between any two neurons is constant. Sampling a neural assembly that contained such structures would likely result in the repeating precise temporal pattens described above. However, such patterns could also be produced by other circuitry. The experimental results and models of HVC support the synfire theory [55, 76, 98], but without comprehensive connectivity data, the existence of synfire chains cannot be proven.

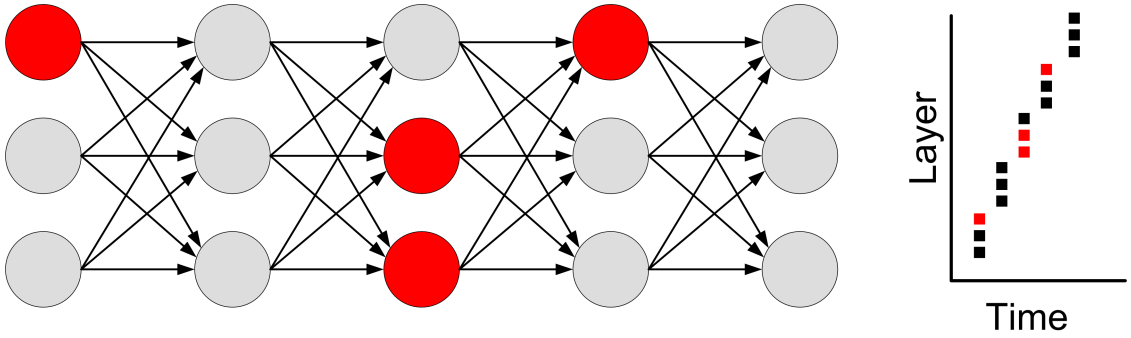


Figure 1.1: Schematic representation of a synfire chain (left) and the neural activity pattern produced (right). Sparse sampling from the structure, as in electrophysiological procedures, would capture only a fraction of the spikes (red) and could result in the precisely timed sequences observed.

Nevertheless, the conjectured importance of synfire chains has inspired a large body of computational work, which has demonstrated the capacity of synfire chains to generate precise sequences of spikes [2, 3, 33]. Extensions include investigations into the role of

such structures in working memory [6, 72, 84], within balanced networks [7, 48, 85, 152], in the presence of noise [69, 159] and with inhibitory modulation [139]. Naturally, the question of development is also of much interest. Early work on the development of synfire chains [13] utilised Hebbian plasticity. More recently, the discovery of spike-timing-dependent plasticity (STDP) provided a new impetus to model the *ab initio* development of synfire chains.

1.3 Spike-Time-Dependent Plasticity

The strongest evidence for the existence of synfire chains comes from song bird HVC. However, HVC development is poorly understood with only a few studies addressing plasticity in this area. A recent gene expression study found that plasticity related genes were expressed less in adult than in juvenile finches [80], which is consistent with the observation that adult finches reproduce the same songs throughout adulthood unless auditory feedback is damaged [91]. Neurogenesis, the addition of new neurons, within HVC is also correlated with the learning and refinement of songs; as song stability increases, neurogenesis decreases [117, 126]. Synaptic plasticity within HVC has not been studied experimentally. However, several computational studies have suggested a possible role for Spike-Time-Dependent Plasticity (STDP) [34, 38, 58, 67, 77, 105] in the development of synfire chains.

Synaptic strength, the effect a presynaptic spike has on the postsynaptic cell (the term *weight* is often used in computational studies) is not static but changes with the activity of the presynaptic and postsynaptic cell. Naturally, there is great interest in activity dependent plasticity as possible mechanism by which learning might occur in the brain. STDP is a specific form of activity dependent plasticity which links the spike-time difference between pre and postsynaptic spikes with the change in synaptic strength. The importance of relative timing when inducing plasticity through stimulation was reported as important in the early eighties [95]. However, it was not until the late nineties when the precise time difference between individual presynaptic and postsynaptic spikes was mapped to the change in synaptic weight [10, 11, 30, 102, 104, 172]. The term STDP was first used to describe this type of plasticity in 2000 [143] and was subsequently reported in a wide range of animal models, brain regions and neuron types [23, 27, 28, 81]. Interestingly, the relationship between the spike-time difference and the change in synaptic strength is different in different preparations (Figure 1.2). To date, the functional relevance of this diversity has not been uncovered.

Despite the diversity in reported STDP rules, theoretical studies have primarily fo-

cused on one particular function. The asymmetrical function in Figure 1.2 A was one of the first [11] and is the most commonly reported [23]. With potentiation when presynaptic spikes precede postsynaptic spikes and depression otherwise, this plasticity rule appears to reinforce causality as originally postulated by Hebb[2]. This STDP rule, which will be referred to as *classical* STDP from here on in, strengthens forward connections and weakens backward connections, consequently, it seems an obvious choice when modelling synfire chain development. Studies which have done just that are described in detail later in this chapter.

The biophysical mechanisms which underlie classical STDP are, to date, only partially understood. However, current evidence suggests that STDP relies on increased calcium concentrations. A brief but large calcium increase causes long term potentiation (LTP), a longer smaller increase causes long term depression (LTD) [23, 50]. Computational models based on these findings, known as "the calcium control hypothesis" frequently predict a second depressive window for long positive spike-time differences; as the calcium falls back from the high LTP producing levels, it must pass through the intermediate LTD producing level before returning to normal. Such a depressive window has been found experimentally in some studies [32, 116, 167]. Augmenting the calcium control hypothesis with additional proposed mechanisms or assuming that synapses are bistable (have only two possible strengths) can remove this second window from the predicted results [50].

The wide variation in STDP rules is not limited to the functions in Figure 1.2; how such functions should be applied is also a matter of much debate. The change in synaptic strength is sometimes modelled as independent of the current synaptic weight (additive) [143] or conversely a function of both the current weight and the STDP function (e.g., multiplicative, power law or something in between) [52, 113, 130, 145, 160]. Studies using additive STDP show weights migrating to either the minimum or maximum values and thus require these limits to be pre-defined [52, 144]. A self-stabilising unimodal distribution of synaptic weights can be achieved if the STDP function is weight-dependent. Specifically, depression must be multiplicative thus maximally reducing large weights and removing the need for a predetermined weight limit [52, 113, 130, 145, 160]. In addition to the weight dependence of STDP rules, in computational models, it is also necessary to define the range of spike pairs included in the calculation. Nearest-neighbour spike pairings take into account only the neighbouring spikes. Alternatively, an all-to-all algorithm uses all spikes (Figure 1.3). The two methods have been shown to give very different re-

[2]"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" [56]

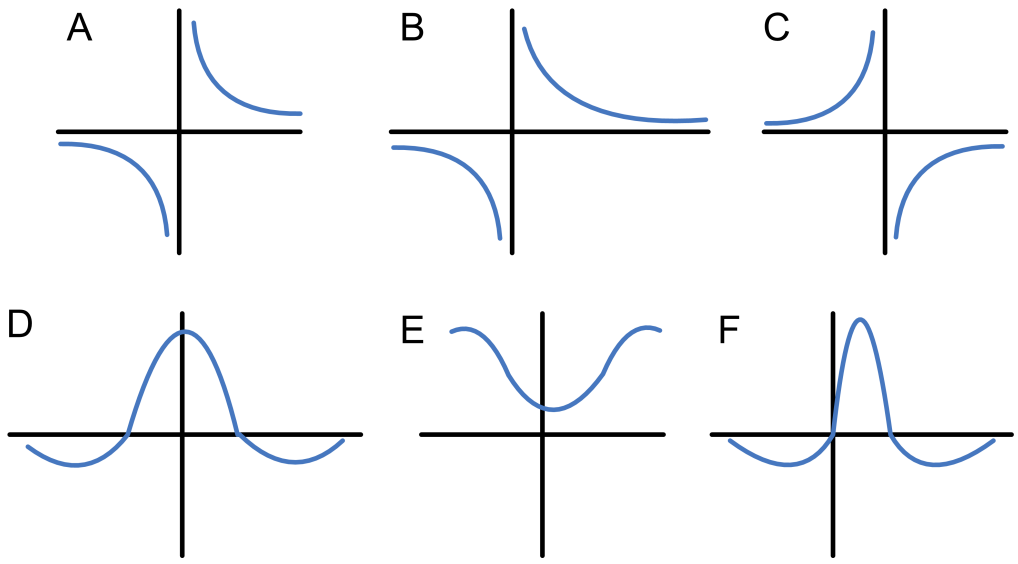


Figure 1.2: A plethora of STDP functions have been reported [1, 23].

sults when applied in simulation [74, 113]. Experimentally, it has been shown that groups of spikes (usually spike triplets) behave in a non-linear way [42, 164]. Next, the existing models of synfire chain development are described.

Figure 1.3: When STDP is applied all-to-all, every spike of A and B is matched and included (left). When applied to nearest-neighbour, plasticity is calculated for adjacent spikes only (right).

1.4 Previous Work

Neural development is modelled at many different levels of detail and using a host of methodologies. The work described here can be broadly categorised as network development models. Each model consists of multiple point neurons connected by synapses which undergo plasticity of some kind. Generally, in a model of network development, the goal is to reproduce and therefore better understand the transition from an original or undeveloped network state to a mature or developed network state. The state of a network can be described either through the activity of its constituent neurons or by the connections between each of those neurons, where usually the former is strongly determined by

the latter.

When studying the development of synfire chains, usually the aim has been to find a plasticity rule or combination of rules that will transform a random or uniform network structure into a feed-forward structure or chain. The goal of the modeller is to show the conditions under which such a transition can occur and to highlight the key determinants and constraints of the model. In doing so, one hopes to provide a testable hypothesis for experimental neuroscience. In this section, the previous attempts to model the development of synfire chains are described. Specifically, the neural model, the synaptic model and the plasticity rules used are all summarised, as well as the initial and final state of the network.

The first work on the development of synfire chains was undertaken by Bienenstock in the early nineties [13]. At the time there had been only one experimental paper published which linked spike-time difference and synaptic plasticity [95] and it was almost a decade before the term spike-timing-dependent plasticity (STDP) had been coined [143]. Bienenstock used a form of Hebbian plasticity in his model, i.e., when a neuron contributes to the firing of another, the synapse between them is potentiated. This was implemented by increasing the synaptic strength if a neuron fires exactly one transmission delay after its presynaptic partner. Neural firing that is not followed (presynaptic) or preceded (postsynaptic) by another spike with the correct transmission delay induces depression. This implementation of Hebbian plasticity actually foretold a type of STDP function discovered experimentally almost two decades later [32, 116, 167]. In addition to the timing based plasticity, Bienenstock includes a mechanism to induce competition between synapses. The total weight of efferent synapses from a single neuron is fixed at a constant value. Similarly, the total weight of afferent synapses is kept constant.

A network of stochastic McCulloch-Pitts neurons is created with either zero or low weighted synapses. The initial activity of this network is therefore infrequent and random, with no propagating activity. Placed within this network is a single (or small number of) seed neuron(s). These seed neurons have initial connections strong enough to cause a group of other neurons to fire. The neurons that receive these connections are referred to as first-generation cells. When a spike in a first generation cell is followed by a random spike in another of the network cells, if the time-difference is correct, the synapse from the first generation cell to the network cell will be potentiated. If this happens a sufficient number of times, the network cell becomes a second generation cell (i.e., reliably fires after the first generation cells). The number of cells that can become second generation cells is determined by the competition component of the plasticity rule. As there is a fixed combined weight for efferent and afferent synapses, once the optimum number of cells

have been recruited, no further cells will be added. As variable transmission delays exist within the network, the resulting structure does not have perfectly synchronised layers as predicted by the synfire chain model [2], hence the result is coined a synfire-braid.

Following the discovery of STDP, this work was revisited and described somewhat more thoroughly by Doursat and Bienenstock (2006) [34]. The main difference in this later work is that the delays are homogeneous, so we are indeed referring to the development of synfire chains and not braids. Here they refer to the plasticity rule (essentially equivalent to the above) as "a simplification of STDP" as define the change in weight of the synapse between cells i and j, w_{ij} as,

$$w_{ij}(t) = w_{ij}(t-1) + b_{ij}(t),$$

with $b_{ij}(t) = +\alpha$ for each $j \neq i$ such that:

$$x_i(t-\tau_{ij}) = x_j(t) = 1,$$

and $b_{ij}(t) = -\beta$ if:

$$x_i(t-\tau_{ij}) \neq x_j(t), \tag{1.1}$$

where α and β are constants of the order 0.1 and 0.01 respectively and τ_{ij} is the transmission delay between neurons i and j. So that, if a postsynaptic neuron fires exactly one delay after the presynaptic neuron the synapse is facilitated. Synapses are depressed in all other conditions.

Again, the network consists of probabilistic neurons where the probability of activation ($x_j = 1$) of neuron j is given by,

$$P[x_j(t) = 1] = \sigma_T(V_j(t) - \theta_j),$$

where

$$V_j(t) = \sum_i w_{ij}(t) x_i(t-\tau_{ij})$$

is the membrane potential of j at time t; θ_j its firing threshold; τ_{ij} is the transmission delay and σ is a sigmoidal function defined as,

$$\sigma_T(V) = \frac{1}{1+e^{\frac{-V}{T}}}, \tag{1.2}$$

where T is a temperature parameter that sets the amount of noise in the system. The result of this probabilistic neural model is a small amount of spontaneous activity in the initial

network.

Again, the restriction on synapse strength is imposed. Here the total weight of incoming synapses must equal the total weight of outgoing synapses and remain at a predefined constant,

$$\sum_i w_{ij}(t) = \sum_j w_{ij}(t) = s_o. \tag{1.3}$$

In addition there is a cap on any individual synapse at weight = 1.

As before, the model results a network structure which grows consecutive layers, the layers grow in width until the limit s_o is reached. The layers continue to be added to the end of the network until N neurons have been recruited.

Restrictions on the total efferent or afferent synaptic weight are reported experimentally [40, 41, 61, 129] and are utilised as a competitive tool more recently by Fiete and colleagues (2010) [38]. Here this restriction is referred to as "heterosynaptic competition". Heterosynaptic competition is used in combination with the now well established classical STDP rule [11], this double exponential having become the primary rule implemented in computational models.

Here the authors demonstrate in two different neural models how classical STDP paired with heterosynaptic plasticity is capable of producing synfire chains. The first model uses an instantaneous binary neuron where neurons are either active $x_i = 1$ or inactive $x_i = 0$. The activity of the neuron is defined as:

$$x_i(t) = \Theta(I_i^E(t-1) + I^{I\text{glob}}(t-1) + I_i^{\text{ada}}(t-1)) \tag{1.4}$$

where Θ is the Heaviside function, $I_i^E = \sum_j W_{ij}x_j + W_0 b_i$ is the summed excitatory drive to the neuron, with W_{ij} the strength of the connection from neuron i to j, W_0 is the strength of the feed-forward input (equal for all neurons), and $b_i \in \{0,1\}$ is the external input. Global inhibition is given by $I^{I\text{glob}} = -\beta \sum_j x_j$, and adaption is modelled as a threshold dependent on past activity, $I_i^{\text{ada}} = -\alpha y_i$, where y_i is a linearly low-pass-filtered version of x_i, with time constant τ_{ada}. α is the adaptation strength. The external input is randomly assigned to neurons in the network so that spontaneous activity is displayed in the initial network state. There is no input or seed group in this first simple case.

The synaptic plasticity rule is given as:

$$\Delta_{ij}^{\text{STDP}}(t) = \left(\frac{W_{ij}}{W_{\max}} + 0.001\right)\left[x_i(t)K(0)x_j(t) + \sum_{\tau=0}^{t} x_i(t)K(\tau)x_j(t-\tau) - x_i(t-\tau)K(\tau)x_j(t)\right], \tag{1.5}$$

where for binary neurons $x_i(t) = 1$, t is an integer time index and τ_{STDP} is in units of

burst durations. In the initial simulations $K(t) = 1$ for $t = 1$ and 0 otherwise. Competition between synapses is imposed through heterosynaptic competition given as,

$$W_{ij}(t) = W_{ij}(t-1) + \eta \Delta_{ij}^{\mathrm{STDP}}(t) - \varepsilon \eta \theta_{i*}(t) - \varepsilon \eta \theta_{j*}(t), \tag{1.6}$$

restraining each weight within $[0, W_{\max}]$. θ_{*i} represents the competitive heterosynaptic plasticity at incoming synapses and θ_{*j} at outgoing. For the summed-weight limit, $\theta_{i*} = \max\left(0, \sum_{\mathrm{k}}(\mathrm{W_{ik}} + \Delta_{\mathrm{ik}}^{\mathrm{STDP}}) - \mathrm{W_{max}}\right)$ and $\theta_{j*} = \max\left(0, \sum_{\mathrm{k}}(\mathrm{W_{ki}} + \Delta_{\mathrm{ki}}^{\mathrm{STDP}}) - \mathrm{W_{max}}\right)$. For the weight growth limit $\theta_{i*} = \sum_k W_{ik}\Theta(\Delta_{ik}^{\mathrm{STDP}})$, and $\theta_{j*} = \sum_k W_{ki}\Theta(\Delta_{ki}^{\mathrm{STDP}})$. Where Θ is the Heaviside function. Self connections are set at zero; η and ε are the learning rate and the strength of the heterosynaptic constraint respectively.

The network produced by this model displays sparse spontaneous activity in its initial state and precise temporal sequences in its final state. Multiple chains are embedded in the final network, the chains are constructed from single neurons each projecting onto the next, hence, a "layer" in these chains is just one neuron. During development the chains grow, with neurons being added to either end. Usually there exits one long chain and many significantly shorter ones.

Changing the initial activity allows the authors to grow chains with width greater than one. As well as the spontaneous activity a set of input or seed neurons are forced to fire synchronously. When this additional activity is included in the model synfire chains with consecutive layers are grown. The size of the layers can be modified by the total synaptic weight parameter. The width of the layer will be the total allowed weight divided by the total individual weight.

In addition to the simple binary model, the authors also describe a conductance based neural model in which spikes and bursts are modelled explicitly. In this model they find that for chains to be robust, multiplicative STDP must be employed. Otherwise, the results confirmed those described above for the binary model.

The studies described so far have all included competition between synapses by limiting the total weight of efferent and afferent connections. Other authors have achieved similar effects through various implementations of 'pruning'. Classically, pruning refers to the retardation of extraneous synapses, neurites and even whole neurons. Pruning is widespread during development where it is often the case that there is an initial over production of neurons and synapses [36, 78, 100, 106, 137, 148, 169]. Two groups have included pruning in models of synfire development.

In a series of papers [64–68] Iglesias and colleagues describe a model in which 10,000 leaky integrate and fire (IAF) neuromimes (80% excitatory and 20% inhibitory) are placed

on a lattice. The membrane potential of the ith unit, $V_i(t)$, is given by

$$V_i(t+1) = V_{\text{rest[q]}} + B_i(t) + (1 - S_i(t))(V_i(t) - V_{\text{rest[q]}})k_{\text{mem[q]}} + \sum_j w_{ij}(t) \tag{1.7}$$

where $V_{\text{rest[q]}}$ is the resting potential for units of type q, B_i is the background activity arriving at i, $S_i(t)$ is the state of the unit $\{0,1\}$, $k_{\text{mem[k]}}$ is the leak conductance. The networks are initially sparsely connected, the connections are placed randomly with probability of connections determined by the distance between cells. Excitatory cells are more likely to make local connections, with long range connections preferred for inhibitory cells.

Excitatory to excitatory connections are plastic and modified according to a classical STDP based plasticity rule. A continuous variable, $L_{ji}(t)$ is potentiated or depressed in line with an additive application of the classical STDP rule. In addition, L_{ji} decays exponentially, thus low activity leads to a silencing of the synapse. For application of the update rule (i.e., when calculating the effect that synapses have on downstream neurons) $L_{ji}(t)$ is converted into a discrete value $A_{ji}(t)$, A = 0,1,2,4. If A_{ji} reaches 0, the synapse is permanently removed, i.e., the lowest value synapses are pruned. In the later works [67, 68] there is also a period of cell death. Before the STDP rule is switched on those cells with no excitatory inputs are removed as are those with firing rates above a threshold.

The initial network activity is driven by external input, this low level firing received by all neurons results in spontaneous activity. Additionally, a subset of neurons are driven to fire according to predefined patterns. In the earliest work [64], the input neurons are spatially arranged so that the input activity appears to pass over the network in a wave. In the following work the inputs are co-located and fire simultaneously.

When the input is distributed over the network, feed-forward structures emerge from within the network, with backwards connections being removed. When the input is fixed, the author finds repeating temporal patterns within the final network activity. It could be said that this work was not entirely successful in growing synfire chains, even with both types of pruning and sparse initial connectivity, stability in the network was not achieved and the temporal patterns were often based on a single unit firing many times.

The second paper to include pruning was by Jun and Jin in 2007 [77]. Here a network of 1000 leaky integrate and fire neurons is created. The neuron's membrane voltage is described by

$$\tau_m \frac{dV_m}{dt} = (E_l - V_m(t)) - g_{\text{exc}}(t)V_m(t) + g_{\text{inh}}(t)(E_{\text{inh}} - V_m(t)) \tag{1.8}$$

where τ_m is the membrane time constant; Vm is the membrane potential; E_l is the leak

reversal potential; $g_{\text{exc}}(t)$ is the excitatory conductance; $g_{\text{inh}}(t)$ is the inhibitory conductance and E_{inh} is the reversal potential of the inhibitory synapses.

The network is completely recurrently connected, excluding self connections. However, 90 % of the synapses are functionally silent (they have a weight of 0). The remaining 10 percent are set to a low weight ($g_{\text{exc}}(0) = g_{\text{init}}$. During the simulation, the weight of the synapse changes in line with the STDP rules in equation (1.9) for potentiation and (1.10) for depression. Note that the rule for depression is multiplicative, while potentiation is additive (G_{LTP} is a constant). As well as STDP, synaptic depression is included, with every synapse depressed at each trial by a tiny amount, $\beta << 1$.

$$
\begin{aligned}
G_{km} &\rightarrow G_{km} + A_{\text{LTP}} G_{\text{LTP}} \sum_{i}^{\text{All Spikes}} P(t_m - t_k^i) \\
P(\Delta t) &\rightarrow \begin{cases} \Delta t/(5) & \text{if} \quad \Delta t \leq 5\text{ms} \\ \exp(-(\Delta t - 5)/\tau_{\text{LTP}} & \text{if} \quad \Delta t > 5\text{ms}, \end{cases}
\end{aligned}
\tag{1.9}
$$

$$
\begin{aligned}
G_{mn} &\rightarrow G_{mn} + A_{\text{LTD}} G_{mn} \sum_{i}^{\text{All Spikes}} D(t_m - t_n^i) \\
D(\Delta t) &\rightarrow \begin{cases} \Delta t/(5.25) & \text{if} \quad \Delta t \leq 5.25\text{ms} \\ \exp(-(\Delta t - 5.25)/\tau_{\text{LTP}} & \text{if} \quad \Delta t > 5.25\text{ms}, \end{cases}
\end{aligned}
\tag{1.10}
$$

where τ_{LTP} and τ_{LTD} are the time constants for potentiation and depression respectively. Synapses can be in one of four states: active, silent, super or withdrawn. Synapses become functionally silent if their weight falls below a threshold, Θ_A, when silent they have no effect on downstream neurons. However, the weight is remembered and modified according to the STDP rules so it may become active again in the future. Synapses become super if their weight rises above a further threshold, Θ_S. There is a limit in the number of super synapses a neuron can have. When this limit is reached, all other synapses are withdrawn. The withdrawn synapses have a weight of 0 and they do not undergo STDP. It is this withdrawal of the non-super synapses that they refer to as pruning and induces competition between the efferent synapses of the same neuron.

To balance the activity of the network, a global inhibitory "interneuron" generates an inhibitory postsynaptic potential in all network neurons in response to each excitatory spike. Each network neuron also receives inhibitory and excitatory background activity resulting in intermittent spontaneous activity (at approximately 0.2 Hz). Finally, a subset of neurons are classified as training nodes and receive excitatory input at the beginning of

each trial.

The results of this model are similar to those already described. In the initial network, the training nodes fire regularly and some of these spikes occur coincidentally just before a network neuron spikes. The resulting spike-time difference insures potentiation of the synapse from the training node to the network node. As a result of many such potentiations, the network neuron starts the first layer, spiking reliably after the training node. And so on and so forth until a chain of around 70 layers is built. At this point backward connections begin to form as the STDP rule is negligible at this spike-time difference.

Here the competition between synapses has been introduced with pruning instead of heterosynaptic plasticity, with the number of strong "super" synapses limited instead of a limit on the summed synaptic weight. The effect is similar; synapses that are part of an active chain will potentiate each time the chain is active. Therefore, synapses will be pushed to the maximum allowed value. Setting a limit either on the total weight or the quantity of efferent synapses will result in a fixed width for the chain.

In the work described so far, competition between synapses is included through plasticity rules (or restrictions on plasticity rules) which operate at the level of individual neurons, termed either heterosynaptic competition or pruning the effect is the same, there is a limit on the number of synaptic partners a neuron can have. Another way to limit the number of synaptic partners is to limit the initial connectivity. Two further authors have successfully grown synfire chains this way.

In two recent papers Masuda and colleagues [105, 151] describe a network which consists of coupled phase oscillators. The dynamics of the oscillators are given by

$$\frac{d\phi_i}{dt} = \omega_i + \frac{1}{\langle k \rangle} \sum_i g_{ji} \sin(\phi_{\mathrm{j}} - \phi_{\mathrm{i}}) + \sigma \xi_{\mathrm{i}} \tag{1.11}$$

where ω_i is the inherent frequency of neuron i, $\langle k \rangle$ is the average in-degree, g_{ij} is a synaptic weight and ξ_i is a white Gaussian noise term (only included in the later paper). Spike times are defined as the time at which ϕ_i crosses 0 in the positive direction. STDP is of the classic exponential form,

$$\Delta g_{ij} = \begin{cases} A^+ \exp(\frac{t_i - t_j}{\tau}), & t_j - t_i > 0 \\ -A^- \exp(-\frac{t_i - t_j}{\tau}), & t_j - t_i < 0. \end{cases} \tag{1.12}$$

The authors find that, in the right parameter range, the initially sparsely and randomly connected network will develop into a feed-forward network. If the network is constructed from oscillating units with a range of intrinsic frequencies, the fastest will become the

pacemaker and lead the group. The "layers" in the chains presented here exist because the initial connection matrix is sparse; the length of the chains is equal to the shortest initial path through the network. If the initial networks had been fully connected, the final state would be synchronous firing.

In contrast to the work described so far, Kitano *et al* [83] presented a model for growing synfire chains with no restriction on synaptic partners. Here the development is driven by the general (spontaneous) activity of the initial network i.e., there is no input or training group. The neural model is given by

$$C_m \frac{dV}{dt} = -g_L(V - E_L) - I_{Na} - I_K - \sum_j I_{\text{syn,j}} - I_{\text{app}} - I_{\text{noise}} \tag{1.13}$$

where I_{noise} gives initial spontaneous activity at 0.5-1.5Hz. Plasticity follows the classical additive form (Equation 1.12)).

The chain described is actually a loop consisting of three groups or layers. The cyclic activity pattern is the key to the maintenance of this structure. The 'backward' connection from the last to the first group is maintained because although both positive and negative spike-time differences occur, the positive spike-time difference is shorter so potentiation dominates. As the authors note, to grow longer chains, there would need to be a restriction on network connectivity.

In the work described so far, the transition has been from random or uniformly connected networks to a either a single feed-forward network or a separation into a few distinct feed-forward networks (typically one large and a few smaller). However, Izhikevich (2004 and 2006) [73, 75] reports the growth of many overlapping synfire braids in one network. Like the pioneering work by Bienenstock [13], the network is connected with a distribution of delays, thus chains with regularly spaced groups or layers are not developed. In this work relatively small synfire braids are developed which give rise to repeating spatiotemporal spike patterns. In addition to heterogeneous delays within the network, the neurons are heterogeneous. The neural model is given as,

$$\begin{aligned} \frac{dv}{dt} &= 0.04v^2 + 5v + 140 - u - I_{\text{syn}} \\ \frac{du}{dt} &= (bv - u) \\ \text{if v(t)} = 30\text{mV}, \quad \text{then} &\begin{cases} v & \leftarrow c \\ u & \leftarrow u + d. \end{cases} \end{aligned} \tag{1.14}$$

The parameters a, b, c and d are adjusted to give different intrinsic firing patterns within

the network. The network is sparsely connected with connections randomly assigned. The earlier work contained more biologically inspired constructions, a notion of space, plus long and short range projections. However, both models give rise to the same development trajectories. Small braids grow from the application of classical STDP (Equation (1.12)); although a fraction of the braids are persistent most will appear and subsequently disappear within one simulation.

A further work which has embedded multiple patterns within a network and also found that each neuron could be a part of multiple distinct patterns used STDP as a fine tuning parameter, rather than the main plasticity rule. Liu and Buonomano (2009) [96] are able to grow chains with what they term 'presynaptic-dependent scaling', defined as,

$$W_{ij}^{\tau+1} = W_{ij}^{\tau} + \alpha_W A_j^{\tau}(A_{\text{goal}} - A_i^{\tau})W_{ij}^{\tau} \tag{1.15}$$

where W_{ij}^{τ} is the weight from i to j at trial τ. α_W is the learning rate and A_{goal} is the target activity. A_j^{τ} is the average activity of neuron i at trial τ. In a network of 320 excitatory and 80 inhibitory integrate and fire neurons, 24 excitatory and 12 inhibitory neurons are forced to fire. There is initially no activity in any other of the units. The plasticity rule increases the weight of synapses between active presynaptic cells and postsynaptic cells which fire below their target. This encourages the growth of chains from the input neurons, and is found to produce feed-forward networks when used as the only development mechanism [19]. However, the millisecond precision associated with synfire chains could not be reproduced; the firing of the structures becomes less precise over time. The additional application of classical STDP [96] helped with the fine tuning of the networks, and allows the authors to embed more than one chain within the network. The authors note that for multiple chains to be embedded, recurrent connections must exist within the network. It is not clear whether the intrinsic structure of a single embedded chain contains any recurrence. Indeed as each neuron appears to fire only once after presentation of the input, this seems unlikely. It is clear that STDP alone would not have been sufficient for development in this set up as the initial network does not exhibit spikes upon which STDP is based.

One further paper addresses the development of synfire chains [58], however, only the first layer of a chain is created. In contrast to the work previously described which implants a small group of training or input neurons, in this paper the entire network is subject to a repeating input pattern. The network consists of 1000 leaky integrate and fire

neurons, 800 excitatory, 200 inhibitory. The neuron is defined as:

$$C\frac{dV_j}{dt} = g(V_{rest} - V_j) + I_j(t), \tag{1.16}$$

where C is the membrane capacitance; V_j the membrane potential of the jth neuron; g the membrane resistance; V_{rest} the resting potential and $I_j(t)$ the sum of the recurrent and external synaptic current inputs. Classical STDP (Equation (1.12)) is employed at the excitatory synapses only. Following development though STDP, presentation of the pattern results in synchronous firing within the network The authors propose that this synchronous firing could be the first layer of a synfire chain. They apply classical STDP in both multiplicative and additive form and observe little difference between the two implementations.

Finally, there have also be negative results when attempting to model the development of synfire chains with classical STDP. Morrison *et al* recently [113] described simulations in which tens of thousands of IAF neurons are sparsely connected. Classical STDP is applied multiplicatively for depression and with a power law dependence for potentiation, all to all spike pairing was used. The authors find that in these networks of realistic size, either the network remains in a balanced and stable state but no structure develops, or, the network enters a pathological state with periodic firing rates within which transient regular patterns can be observed. Similar divergent results are explained using a mean field approach [86]. This approach predicts that the number of neurons recruited by the input group is subject to an unstable fixed point. The size of the input group is either below the fixed point and no structure emerges. Or, the size of the input group is above the fixed point and the entire network is recruited by the input. The authors argue that such unstable dynamics will always be obtained when realistic network sizes are combined with weight dependent STDP and that the success of previous models is artifactual due in the main part to their insufficient scale.

The vast number of recent papers devoted to the development of synfire chain structures is evidence of the substantial interest in this subject amongst computational neuroscientists. With overviews of each of the works presented together, it is possible to see that broad range of neural models have been successfully employed to demonstrate the development of synfire chains. This suggests that the spike production mechanisms are inconsequential when considering the key elements of the developmental process considered here. So, which are the elements of development that are essential to grow synfire chains? It is useful to summarise those aspects common to the models described above and therefore likely to contibute to their success.

Each model described here that successfully grew synfire chains has employed a spiking neural model, which is logically a necessity to calculate the spike-time differences for STDP. The majority of the networks are constructed from a combination of inhibitory and excitatory units, either with a small fraction of network units being inhibitory, or by including a single inhibitory interneuron. Activity within the network usually consists of regular firing at a small subset of nodes, often referred to as an input group with the remainder of the network exhibiting sparse random activity at lower rate. Synaptic plasticity at excitatory synapses, with the exception of [13] and [19] depends on the application of classical STDP, the exponentially decaying function that induces potentiation for positive spike-time differences and depression for negative spike-time differences, or a variant there of. Finally, in addition to STDP, a competitive mechanism is included that restricts the number of synaptic partners a neuron can have.

The above summary clearly outlines the current best guess for the development of synfire chains, and in doing so appears to provide a framework for the development of such structures, either experimentally or within modelling studies. To test this framework and explore its limits, a new study based on the above principle was undertaken. The results of this study plus systematic reduction of the model highlighting the necessary components are presented in the next chapter.

Chapter 2

Growing Synfire Chains

The previous chapter outlined the possible importance of synfire chains in neural processing and described the current theories regarding their development. Here, a model incorporating the common elements of those models is described. This model is an attempt to recreate the current knowledge in this area and to test the limitations and boundaries. In doing so, the aim is to identify those parts of the previous models that are truly necessary for the observed development. The reduction of the model to its core components allows deep analysis of the developmental process revealing the key mechanisms at work. The section begins with a model description, then simulation results are presented. Initial results are followed by an investigation into the working parameter ranges. Finally a step by step reduction of the model is given.

2.1 The Model

The model presented here is inspired by that of Jun and Jin (2007) [77], which had been recently published at the time this work was undertaken. This model is not an exact reimplementation: it is scaled down due to hardware restrictions and some parameters have been changed. The neural simulation software NEST [101] (version 1.9.7918) was used to undertake this modelling work. Bespoke neural and synaptic models were written which extended the simulator. These can be found along with the simulation scripts on the accompanying CD.

2.1.1 The Neural Model

The network consists of 100 network neurons and 5 input neurons. Each input neuron has efferent synapses onto every network neuron, but receives no afferent synapses from the network. The network neurons are completely recurrently connected excluding self connections. As in [77], an leaky integrate and fire (LIF) model was chosen for the neural model. The change in membrane potential, V_j, of the neuron j is given by,

$$C\frac{dV_j}{dt} = -g_L(V_j - E_L) - g_{\text{ex}}(V_j - E_{\text{ex}}) - g_{\text{in}}(V_j - E_{\text{in}}) \tag{2.1}$$

where C is the capacitance, E_L, E_{ex} and E_{in} are the reversal potentials of the leak, excitatory and inhibitory synaptic currents respectively, g_L is a leak conductance, and g_{ex} and g_{in} are the synaptic conductances given by,

$$\frac{dg}{dt} = -\frac{g}{\tau_{\text{syn}}} + \sum_{i=1}^{N} W_{ij} s_j(t-d), \tag{2.2}$$

where τ_{syn} is the synaptic time constant, W_{ij} is the weight of the synapse and $s(t)_j \in \{1,0\}$ is 1 if neuron j fired at time t and 0 otherwise, d is the delay between neurons.

If the membrane potential, V_j, reaches the firing threshold, θ, the neuron is said to have spiked and V_j is reset to the reset potential, V_{reset}. The neuron then enters an absolute refractory period of t_{ref} where V_j is fixed at V_{reset}.

In addition to the excitatory neurons, the network contains one inhibitory interneuron neuron which has synapses from all and to all excitatory neurons in the network. This inhibitory neuron fires a spike for each one received. The synapses to and from this neuron remain static.

2.1.1.1 Synaptic Plasticity

The weights of intra-network excitatory synapses, W_{ij}, are modified according to the classical STDP rule (Equation (2.3). In line with Ref. [77] the curve is maximal at small positive spike-time differences (see Figure 2.1). The importance of the linear behaviour around $\Delta t = 0$ is addressed later in this chapter (see Section 2.1.3.2). Unlike previous authors, here the STDP rule is applied between nearest neighbour spikes only (i.e., only the very last/next spike is used in the STDP calculation, not every previous spike). This is computationally more efficient and does not affect the results (see Section 2.1.3.5). The

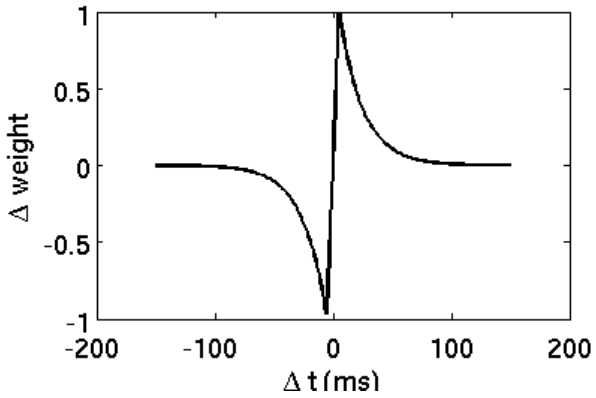


Figure 2.1: The STDP function used in the model.

STDP modification rule is,

$$\begin{aligned}
\frac{dW_{ij}}{dt} =& A_{\mathrm{LTP}} B_{\mathrm{LTP}} P(\Delta t_{ij}) - A_{\mathrm{LTD}} W_{ij} D(\Delta t_{ij}) \\
P(\Delta t_{ij}) \rightarrow & \begin{cases} \Delta t_{ij}/(5) & 0 < \Delta t_{ij} \leq 5\mathrm{ms} \\ \exp(-(\Delta t_{ij}-5)/\tau_{\mathrm{LTP}}) & \Delta t_{ij} > 5\mathrm{ms} \\ 0 & \text{otherwise,} \end{cases} \\
D(\Delta t_{ij}) \rightarrow & \begin{cases} \Delta t_{ij}/(5.25) & 0 > \Delta t_{ij} \geq -5.25\mathrm{ms} \\ \exp(\Delta t_{ij}+5.25/\tau_{\mathrm{LTD}}) & \Delta t_{ij} < -5.25\mathrm{ms} \\ 0 & \text{otherwise,} \end{cases}
\end{aligned} \tag{2.3}$$

where Δt_{ij} is the spike time difference having removed the delay ($\Delta t_{ij} = s_j - s_i - d_{ij}$). Removing the delay from the spike time difference assumes that the delay is entirely axonal, as axonal delay is not included in STDP calculations experimentally. The STDP time constants τ_{LTP} and τ_{LTD} set the rate of decay; $A_{\mathrm{LTP}}, A_{\mathrm{LTD}}$ are the learning rates; B_{LTP} is a constant which scales potentiation only, thus depression is multiplicative and potentiation is additive. The importance of this is also explored later in Section 2.1.3.5.

In addition to STDP, a restriction on the number of strong synapses is imposed, taken directly from Ref. [77]. When a weight rises above a threshold, Θ_S, the synapse is defined as super. A neuron may have only N_{sup} super efferent synapses; there is no restriction on afferent synapses. When the limit is reached, all other efferent synapses are withdrawn. In the withdrawn state the synapse has an effective weight of 0 and therefore has no effect on downstream neurons. However, a memory of the weight remains and it continues to undergo STDP. If one of the super synapses drops below the threshold, Θ_S, the withdrawn synapse are reactivated. Furthermore, any synapse with a weight below Θ_A is also effectively silent, although still undergoing STDP in the same way and will be reactivated if

Table 2.1: Model Parameters

	LIF Simulation Parameters	Binary Simulation Parameters
N	100	100
N_{in}	5	5
N_{sup}	10	10
d	2.0 ms	2 ms
Θ_S	13.5	0.533
Θ_A	6.75	0.133
W_{max}	20.25 nS	0.66
A_{LTD}	0.0105	0.0105
A_{LTP}	0.01	0.01
B_{LTP}	10.125	0.33
$\tau_{\text{syn\_in}}$	2.0 ms	NA
$\tau_{\text{syn\_ex}}$	0.2 ms	NA
θ	-50 mV	1
t_{ref}	25 ms	10 ms
E_L	-85 mV	NA
V_{reset}	-80 mV	NA
V_m	-80 mV	NA
E_{in}	-75 mV	NA
E_{ex}	0 mV	NA
g_L	1.125 nS	NA
C_m	22.5 pF	NA

it moves above the threshold in the future (provided the super synapse limit has not been reached). The simulation parameters can be found in Table 2.1.

2.1.1.2 Spontaneous Activity

Network neurons receive stimulation from the input cells and the other network cells. In addition, network neurons also receive a combination of excitatory and inhibitory background inputs which cause the neurons to fire stochastically. The background activity is modelled as a collection of Poisson inputs with varying weights. Each neuron receives 10 excitatory and 10 inhibitory inputs with rates 4 Hz and 20 Hz respectively. The weights of the excitatory inputs are evenly distributed between 2.6 and 26, the weights of the inhibitory inputs are evenly distributed between -0.2 and -2.0 (weight corresponds to peak conductance in nS at the synapse). Input neurons do not receive background input but are forced to fire periodically (every 300 ms).

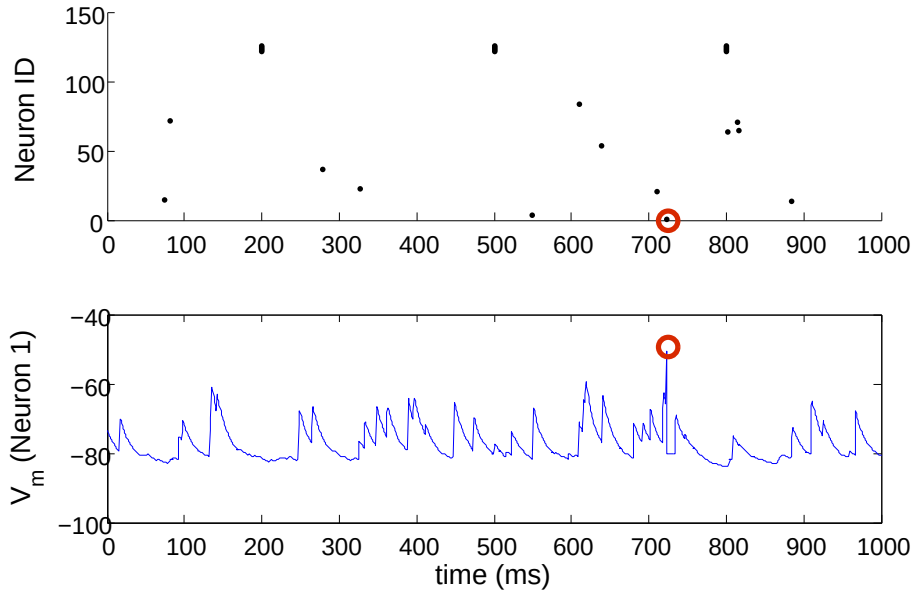


Figure 2.2: The initial network activity in the LIF model (top) consists of sparse spontaneous firing in the network and regular firing in the input neurons. In this model spontaneous activity is driven by fluctuations in membrane potential (bottom) caused by the incoming background activity. The red circles in each highlight a spike generated in neuron one, as its membrane potential crosses threshold.

2.1.2 Results

The final network activity and weight configuration for a typical simulation is illustrated in Figure 2.3 with the network structure presented in 2.4. The process described for previous models in Chapter 1 also occurs within these simulations. The cumulative effect of chance coincident spikes cause the synapses between the inputs and a network node to cross the firing threshold. At this point the network neuron becomes part of the chain. When the input has the prescribed number of postsynaptic partners, no further additions are made to that layer. The limit on the number super synapses, set at ten (five for the input nodes), results in chains of width of either five or six. Each input recruits the same five input nodes, as the input neurons fire synchronously, they undergo STDP synchronously. That is, they will each have exactly the same spike-time difference from any particular network node. Following the recruitment of the first layer, five network nodes now fire at approximately the same time. The variation in spike-times in the first layer, the *jitter*, is a result of the background activity which creates a fluctuating membrane potential (Figure 2.2). As each neuron has a different membrane potential, each neuron will take a very slightly different time to reach threshold following presentation of the input. The net result of these small positive spike-time differences is potentiation between members of the same layer. So the first layer (and similarly for all subsequent layers) becomes

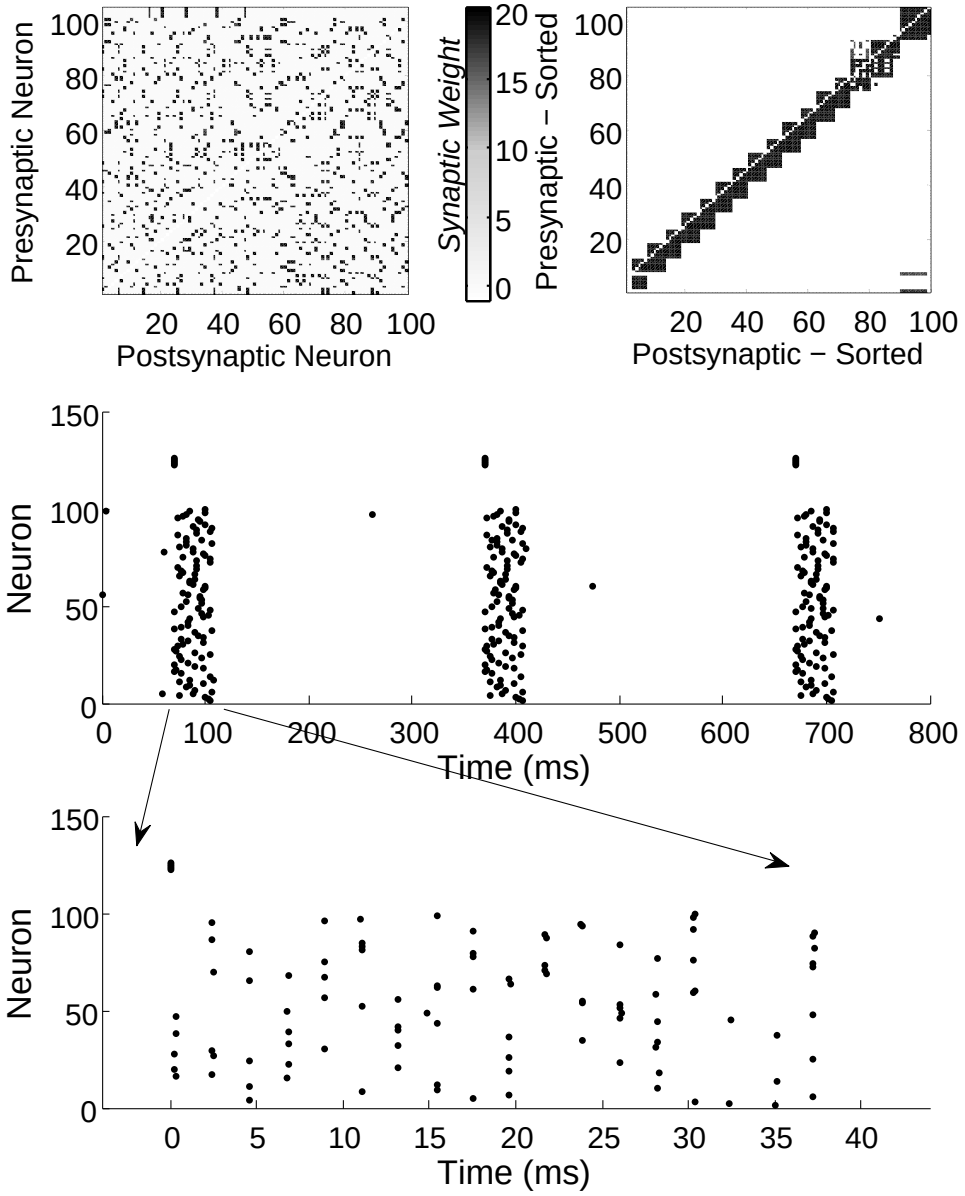


Figure 2.3: The final structure of the network of LIF neurons is displayed, with the neuron IDs sorted by spike time difference from the input (top right), and as originally assigned (top left). The raster plot (middle) clearly shows a repeating structure. The raster plot (bottom) illustrates the layer structure, with spikes occurring in groups around 2 ms (d) apart. Parameters are as in Table 2.1.

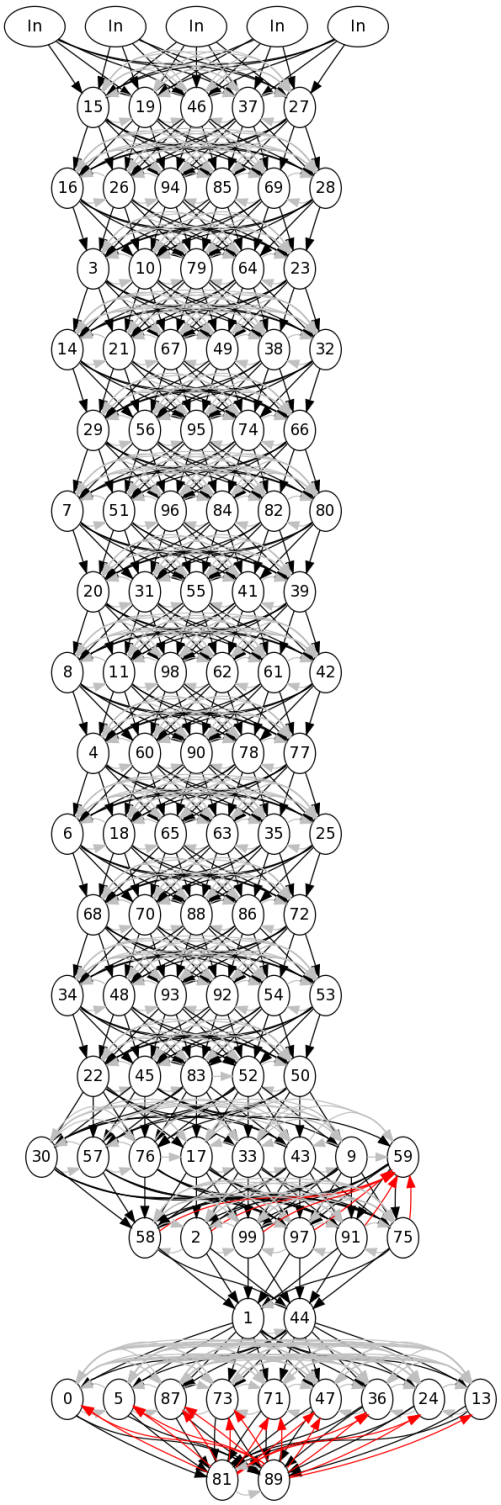
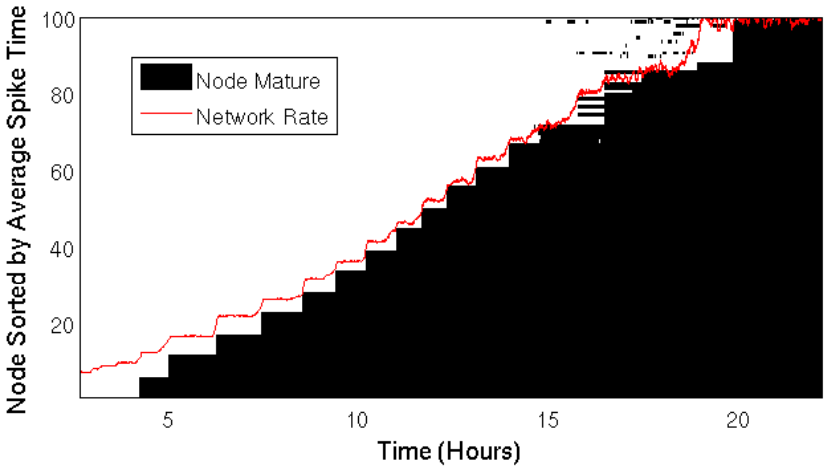
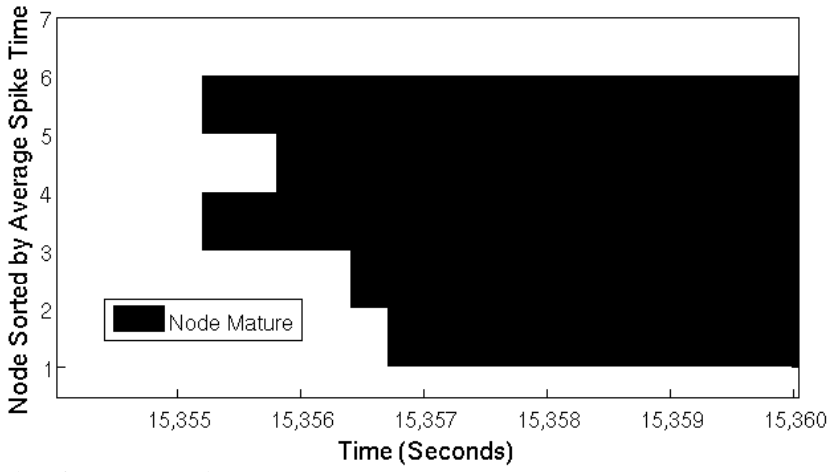


Figure 2.4: The final chain structure in the LIF model. The structure is determined from the weight matrix. Connections are lateral (grey), forward (black) or backward (red).

recurrently connected. As there are always five neurons in the first layer; four of the allocated ten super synapses are used to connect to the other members of the layer, leaving six free for feed-forward connections to the next layer. The next layer is recruited as previously described and will be six neurons wide. These neurons will again form intra-layer connections, this time taking five super synapses, thus leaving five for feed-forward connections. This process continues until the nodes are all recruited, leaving a chain which has alternating layer sizes of five and six. The growth of wider or narrower chains can be induced through modification of the pruning parameter (the number of allowed super synapses).



(a) Entire Simulation



(b) First Layer Only

Figure 2.5: (a) The neurons mature (i,e,. lose all but their strongest ten synapses in layer order, with each layer forming at roughly the same time. As more neurons are added to the chain the network rate increases accordingly. (b)The first layer forms within seconds.

When a neuron has the allotted number of super synapses, its other synapses are withdrawn and the neuron is described as "mature". Figure 2.5 maps the maturation of neurons

throughout the simulation. (The neurons have been sorted by final spike-time so that those with the lowest indexes are in the first layer and those in the last layer have the highest indexes.) It takes almost a day of simulated time to create the whole chain. As illustrated in Figure 2.5(b), the nodes within a single layer will mature within a few seconds of each other. The trajectory of all the efferent weights from one input neuron are in Figure 2.6. Following the individual weight changes over time helps illustrate the general developmental process. As each network neuron has the same rate of spontaneous activity, the probability of potentiation (depression) is equal. However, because the activity is stochastic, each individual synapse's trajectory will differ around the expected value. Input neurons or recently recruited neurons fire at a high rate; therefore their synapses undergo the most plasticity. Before they mature, all the neurons' efferent synapses fluctuate according to the STDP rule, with the net effect being a slow steady increase. When the synapses hit the silence threshold, Θ_S, they become active and continue to increase. When they get strong enough to cause a network neuron to fire, they potentiate with every further presentation of the input and quickly ascend to $W_{\max}$. It is chance effects that cause the 'winning' synapses to become super first, but when enough have done so, the competitive pruning rule kicks in and the rest are withdrawn. As each layer, once formed, has roughly synchronous spike-times, each of their synapses onto any particular network neuron will undergo roughly the same plasticity. Hence, all the neurons of a particular layer will become mature at around the same time, when the final member of the next layer is recruited.

Note that in Figure 2.4 some backwards connections persist in the final layers. Also note that these neurons fall in and out of maturity. This is a result of there being only a few network neurons left for the final layers to recruit. Consequentially, the few remaining neurons form connections with other already recruited neurons, but being backwards connections and contrary to the workings of the STDP rule, these connections will be depressed quickly and removed. This leaves the synapses seeking new post synaptic targets, but with none available that satisfy the conditions for stable recruitment, the circle of random affiliation and then rejection continues. Hence, networks will usually have non-regular layers at the end of the chain.

One question asked by other studies of synfire chain development is whether multiple chains can be embedded in to the same network [38, 96]. This model was tested for that ability by including multiple input groups, a further five neurons were added to the network, as with the original inputs they have synapses onto the network but do not receive synapses either from the network or each other. The result of multiple simulations was that in every case the chains converged, that is, at some point in the chains devel-

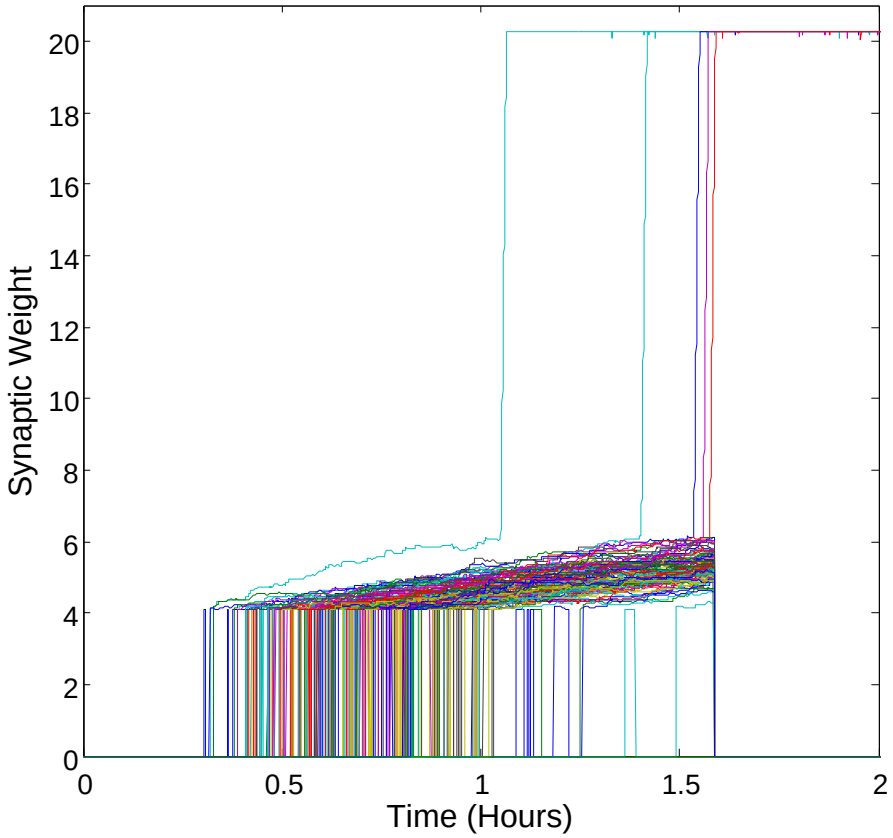


Figure 2.6: The synapses from an input node onto the network. Each line tracks the weight trajectory over time. Note the jump from 0 on hitting the lower threshold (0.3 hours), then the fast rise $W_{\max}$ once the firing threshold has been reached (5 times between 1 and 1.6 hours). The input nodes are allowed 5 supers, once these are allocated, the other synapses withdraw to 0 (1.6 hours)

opment a node is recruited to both chains. When this happens all subsequent recruits (to the same or later layers) are shared by both chains. The result is that the two chains have unique initial layers and shared later layers. Therefore, part of the network activity pattern produced by the inputs is identical (Figures 2.7 and 2.8). Adding a further input to the network only intensifies this effect. Up to five inputs were tested and in most cases chains converged. Occasionally, when many inputs were used, some small chains were maintained with no crossover. This can be attributed to a lack of available network neurons, with only 100 neurons in the network and many inputs it is possible for all neurons to be recruited, without the phenomenon of shared neurons (simultaneously recruited to two chains) happening between each and every input.

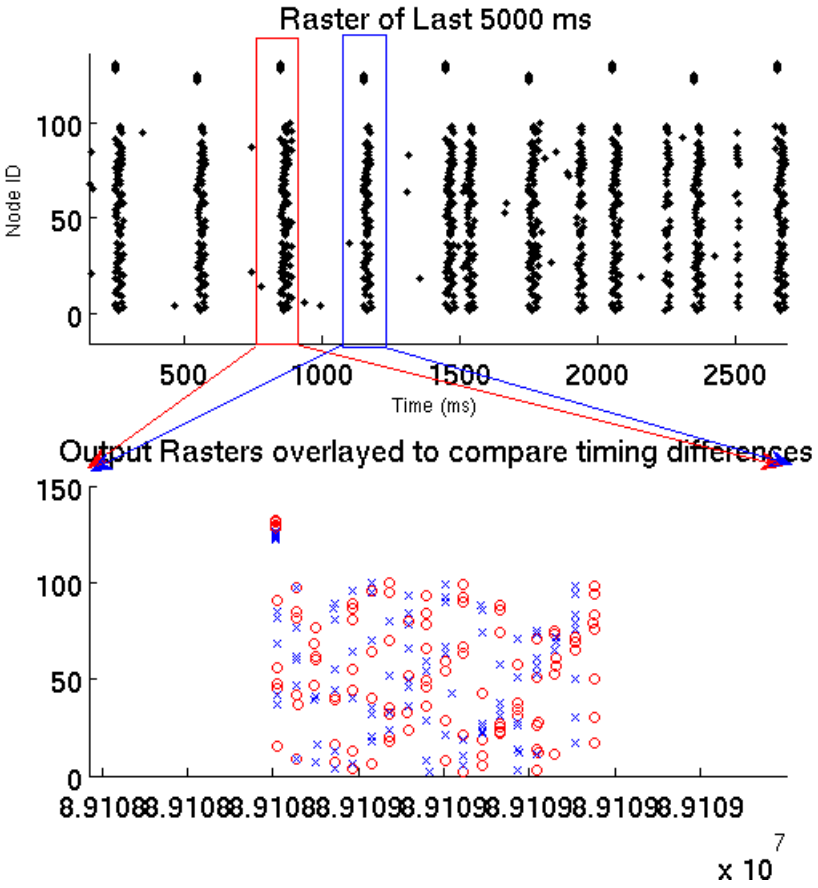


Figure 2.7: The activity of the network shifted by 300ms (to the time of the next input) and overlaid, the repetition of the pattern beginning at layer three in the red and four in the blue.

2.1.3 Reduction

Having established that it is possible to grow synfire chains using the model described here (Eq.s (2.1- 1.9)) and based on previous work in this area, attention now turns to understanding which parts of the model are strictly necessary to observe the required development. Let's first focus on the refractory period of the network.

2.1.3.1 Refractory Period

The model described was based on previous published work [77] and as such parameters were set in accordance with this successful model. However, one parameter stands out as being well outside physiological values. The refractory period of neurons is usually a few milliseconds, but here it is set to 25ms. The reason for this departure from biological values is not clear. Hence an investigation into the effect of reducing the refractory period

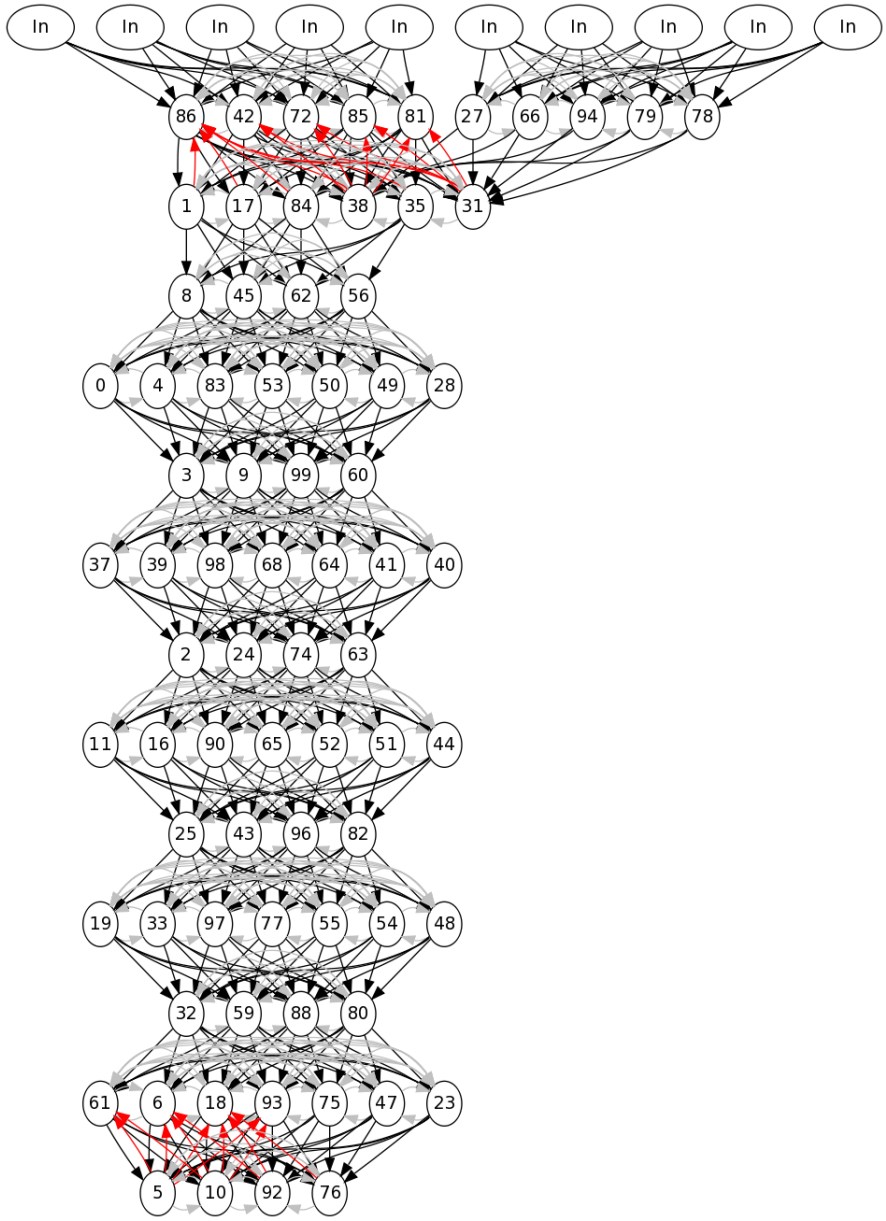


Figure 2.8: A neuron being simultaneously recruited to the first layer of input one and the third layer of input two, causes a merging of the chains. Model described by Eqs ((2.1-1.9) and parameters in Table 2.1 with a addition of an extra input group of five neurons.

was undertaken. Simulations were run with the refractory period between 1 and 20 ms. Only those simulations with a refractory period of 8 ms or greater were successful in growing chains.

Where the refractory period is set to less than 8 ms, chains do not develop. Instead the network develops a very high level of repeating activity (Figure 2.9). Recall that each layer is recurrently connected. When the refractory period is large, those recurrent

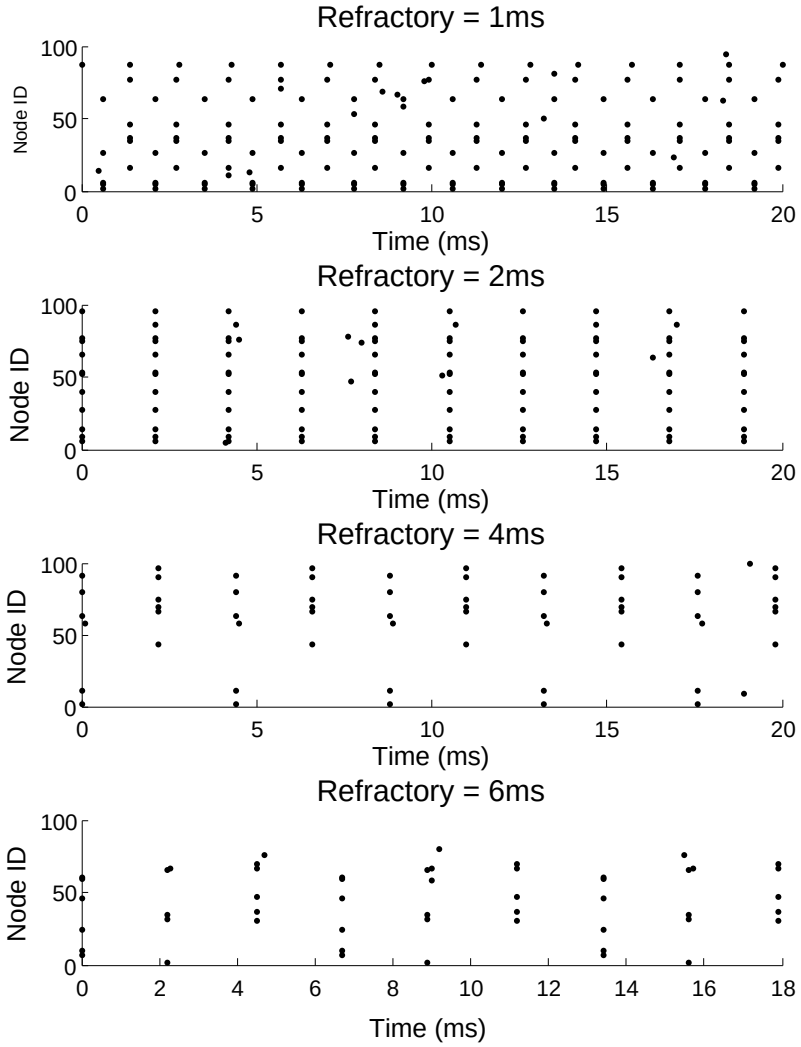


Figure 2.9: Raster plots of simulations that have failed due to short refractory period, which causes recurrent reverberating activity. Note now each spike pattern repeats at exactly the refractory period of that simulation.

connections actually have no effect on the network activity because the spike is received during the refractory period. If the refractory period is smaller than the delay (d) the spikes generated by a layer will be also received at that layer one delay later causing the layer to fire again. This reverberating activity halts chain formation and sends the network into a period of extreme excitation.

This phenomenon is also present for simulations where the refractory period is slightly larger than the delay. Here, two or three layers (depending on the refractory period) can become a loop, similar to that seen in Ref. [83]. Loops of up to three layers will be maintained if allowed to form as the synapses involved are in a balanced state undergoing

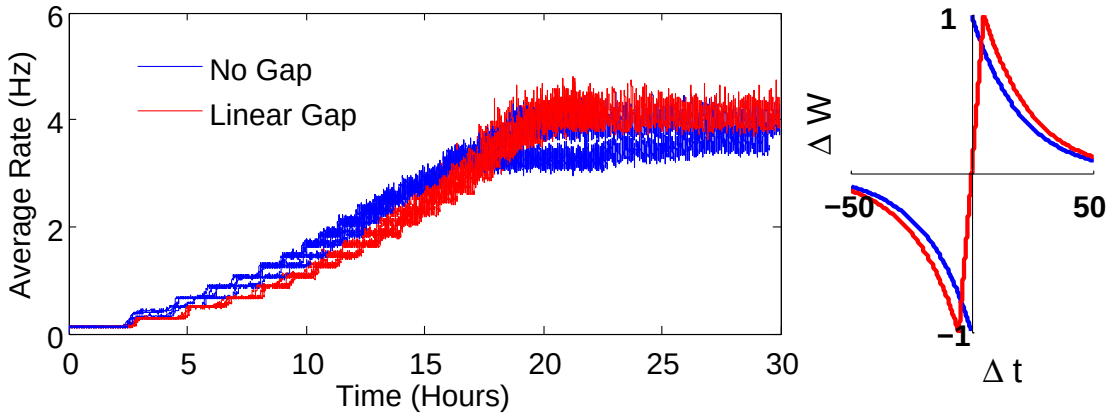


Figure 2.10: The average rate of neurons within the network (left). Five separate simulations are plotted for each rule. The rate is averaged over all neurons for five minutes of simulated time. The STDP rule with a linear gap (red) and the rule without (blue).

both potentiation and depression during the loop thus maintaining the structure.

It should be possible to address these issues of recurrence and looping activity through compensating mechanisms. One possible mechanism could be stronger fast inhibition within the system, structured so that cells that have recently fired receive inhibitory post-synaptic potentials which counteract any excitatory potentials and inhibit the cell's activity. Another way of controlling the activity of the neuron without resorting to long refractory times, is to manipulate the time constant of the excitatory potential. Long time constants can lead to bursting behaviour. Lastly, the STDP rule itself can be modified so that loops, when they occur, are removed. To do this, the balance of depression to potentiation must be set correctly, (depression must be higher at large synaptic weights) thus breaking the loop.

It was beyond the scope of this study to thoroughly investigate the above solutions, hence simulations continue with refractory periods of at least 8 ms duration.

2.1.3.2 STDP Rule - Behaviour Around Zero

The STDP rule used thus far (Equation 2.3) was taken from [77] and included a linear region between -5.25 ms and 5 ms. Whilst it is not known exactly what behaviour occurs at 0, most modellers place no plasticity at this point. Including decay around 0 is unusual. Because of this, and the very slightly different behaviour for small negative and positive time differences, it is appropriate to ensure that these choices are not necessary for development. If exactly these values are essential for chain development, one could not consider the system robust.

Two versions of classical STDP are shown in Figure 2.10. Both successfully generate

chains of the type previously described. The only slight difference is that in the absence of a linear regime, networks develop slightly faster (see Figure 2.10). The explanation for this expedited development lies in the value of the STDP function at very small spike-time differences. Once a neuron is recruited, most of the potentiation steps will be a result of spike transmission from the previous layer (or input), the duration of which will always be small, hence larger potentiation steps and faster development are observed for the more frequently applied function without the linear gap.

The slight increase in development speed is very minor and does not constitute a qualitative difference in the results of the model. As such, it is possible to say that the development of synfire chains in our model and in Ref. [77] does not rest upon this precise aspect of the STDP function.

2.1.3.3 Neural Model

The integrate and fire model used in the above investigations is a popular model within computational neuroscience as it captures the basic time dependent dynamics of neurons; aggregating inhibitory and excitatory inputs plus a leak current. Neither the space dependent aspects of neural processing or voltage dependent ion channels are explicitly modelled. Making these abstractions saves computational power/time and removes often unnecessary, or worse distracting, levels of complexity. In some cases this would be an over simplification (dendrites are very important in some specific neural types, those which perform coincidence detection for example [29]). However, for a wide range of neurons, very simple models can recreate the spiking response to complex inputs accurately and efficiently [47]. Here, because we are primarily concerned with the state and evolution of the synapses and not the neural (spiking) dynamics, it may be appropriate to reduce the neural model from a LIF to a binary model. At the moment there are over twenty free parameters in the system. If a simpler neural model can be used and still achieve the developmental trajectory of interest, then this would reduce the number of free parameters. Leaving only those parameters which determine the behaviour we are studying, the development of synfire chains.

To address the question of the importance of neural dynamics in synfire chain growth, the integrate and fire neuron in the above model is modified towards a simple binary neuron by assuming infinite leak. The new model has no memory of synaptic or any other events (hence infinite leak). It simply sums the synaptic inputs at the current time step and is described as either firing $\{S = 1\}$ or not $\{S = 0\}$. Whilst not strictly necessary, it is usual to pair such a simple neural model with a simple synaptic function, here we model the synapses as delta functions. A synaptic input is instantaneous (i.e., happens at exactly

spike-time plus delay and has infinitely small duration). All synchronous synaptic events are summed to calculate the neuron's state. The membrane potential, V_j (with arbitrary units) of the postsynaptic cell j is increased by exactly the weight of the synapse (with the same arbitrary units) from the presynaptic cell i at time $t_i + d$ where t_i is the spike-time in i and d is the delay.

$$\begin{aligned} V_j(t) &= E_L + \sum_i S_i(t-d)W_{ij} \\ S_j(t) &= \Theta\left(V_j - \theta\right) \end{aligned} \tag{2.4}$$

where Θ is the Heaviside function (0 for negative values, 1 otherwise) and θ is the firing threshold. As before, after a spike the neuron enters an absolute refractory period of t_{ref}.

In the previous model, spontaneous activity was a product of many excitatory and inhibitory inputs causing fluctuating membrane potential. Clearly, with the memoryless dynamics created here, this approach will not cause the neuron to fire. Instead the neuron is forced to demonstrate sparse random activity through a single excitatory input with a weight high enough such that a single spike from the input causes the network neuron to fire. Figure 2.11 illustrates the different subthreshold dynamics for this model. Note the difference to Figure 2.2.

Allowing the simulation to run its course, we see similar development of layers, one by one with each layer consisting of five or six neurons. The final synaptic weights and network activity are shown in Figure 2.12. Note that the final network appears to be structured in exactly the same way as when the full integrate and fire model was used (Figure 2.3). The one exception is the jitter in spike time amongst members of the same layer. In the new binary model the spikes in any particular layer are exactly synchronous. This is a natural consequence of the change in the way the spontaneous activity is produced. In the previous model, fluctuations in membrane potential caused both the spontaneous activity and the jitter within a layer. Here no such fluctuations exist and as the current neuron model has infinite leak, the synchronised spiking is necessary for activity propagation; spikes must happen at exactly the same time to be combined in the receiving neuron.

These simulations have shown that synfire chains can be grown with the simplest of binary neuron models and that the dynamics of spike production are not important to the development trajectory of interest here. The trade off for the simpler model is however, less realistic activity patterns, i.e., synchronised firing amongst neurons belonging to the same layer. This now gives us a simple test model in which the importance of the remaining parameters can be addressed.

Having reduced the neural model from integrate and fire to a simple binary neuron,

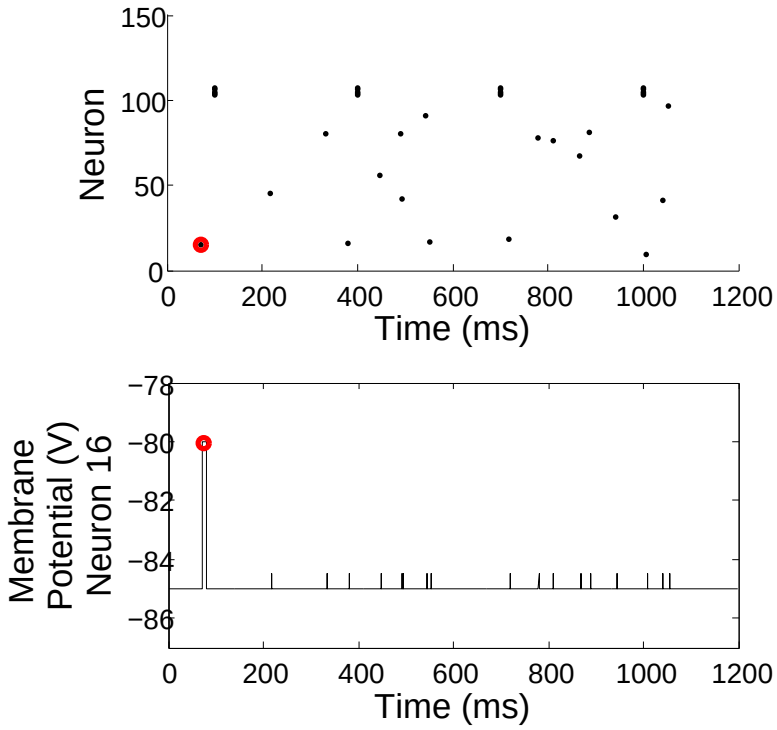


Figure 2.11: Raster of initial network activity (top). Membrane potential of Neuron 16 (Bottom). Red circles mark spike event in neuron 16 in both plots. Note that the membrane potential remains at threshold throughout refractory period, then returns, to the resting potential. The smaller jumps in voltage are the result of other network spikes. The intra-network weights are as yet too small to induce supra-threshold activity

the power of the NEST simulation platform is no longer required. Given the simplicity of the new model, writing a new simulator in python allows faster experimentation. The new neural model is described by Equation 2.4. For simplicity the parameter values are scaled so that membrane potential is in the range $\{0,1\}$ (see Table 2.1). STDP follows the usual STDP function (Equation 1.12 and blue curve in Figure 2.10). This new implementation gives results which are qualitatively the same (Figure 2.13). Unless stated otherwise, all experiments in the remainder of this chapter used the binary model.

2.1.3.4 Spontaneous and Input Activity

One of the key components of all the models described in this chapter was the interaction between sparse random spontaneous activity within the unstructured network and regular higher frequency input activity. Here the importance of the relative frequencies of the two

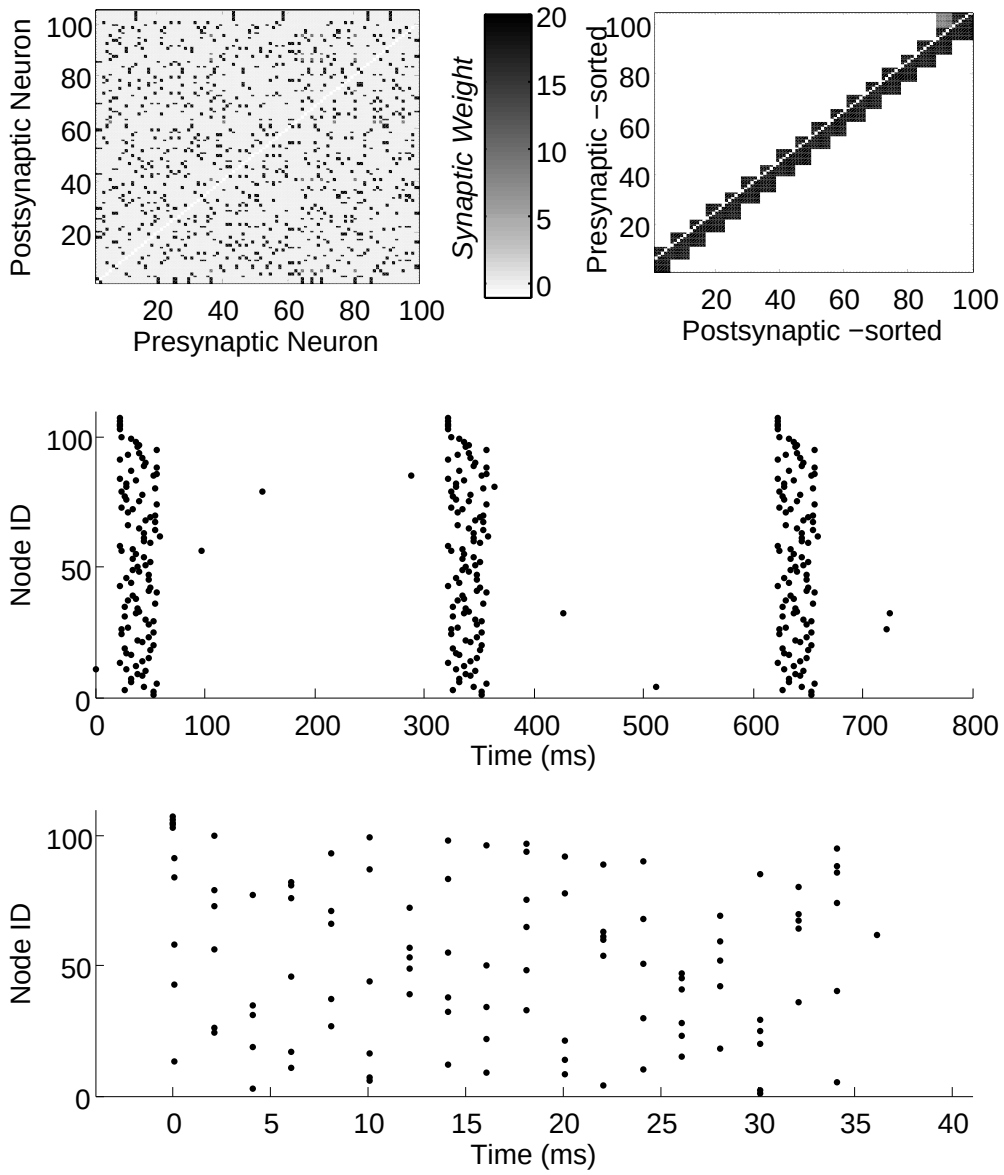
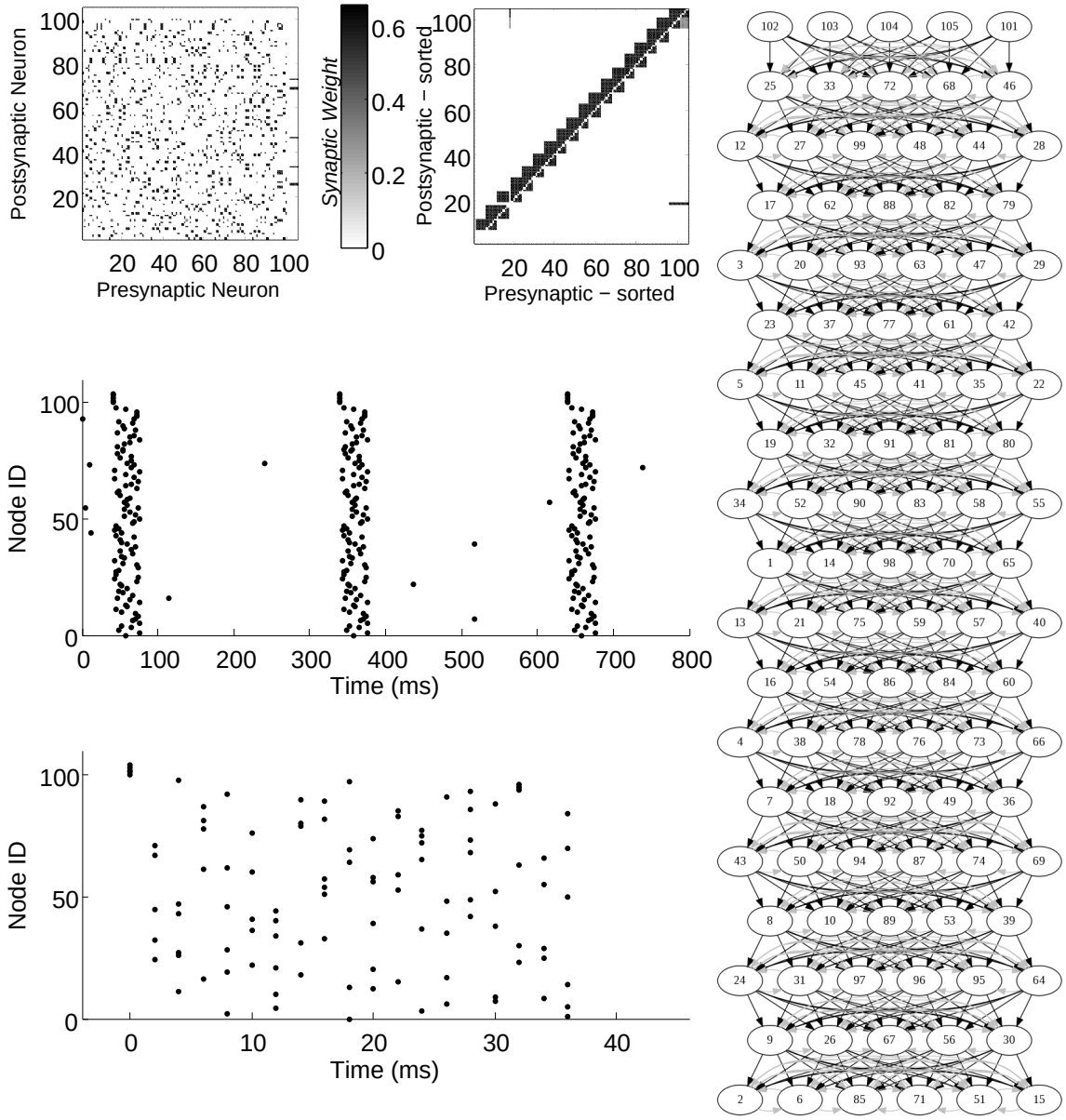


Figure 2.12: The final structure of the network is displayed, with the neuron IDs sorted by spike time difference from the input (top right), and as originally assigned (top left). The raster plot (middle) clearly shows a repeating structure. The raster plot (bottom) illustrates the layer structure, with spikes occurring in groups exactly 2 ms (d) apart.



(a) Network Activity and Weights

(b) Network Structure

Figure 2.13: The final activity of the network and structure of the resulting chains are not altered by reducing the leaky integrate to the binary neural model

types of activity are investigated.

Simulations were run with spontaneous rates between 0.1 Hz and 51.2 Hz and input rates between 0.625 Hz and 10 Hz. Chains only develop in a narrow range of cases. Figure 2.14 shows the final network weights sorted by spike time. Those simulations which successfully developed chains have low spontaneous rates and significantly higher input rates. In simulations with high spontaneous rates, the network neurons begin to form connections between each other and the resulting structures contain recurrent connections

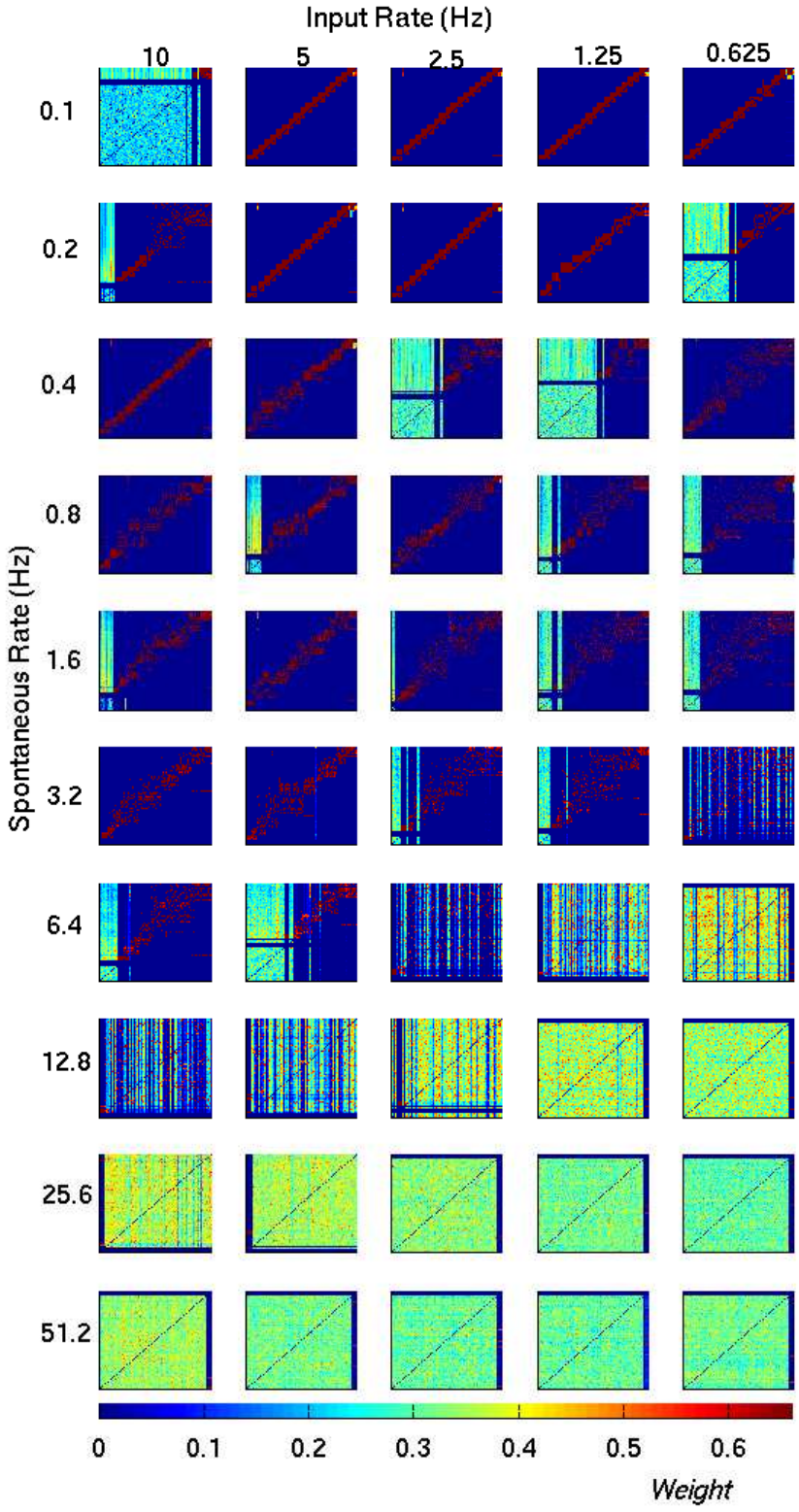
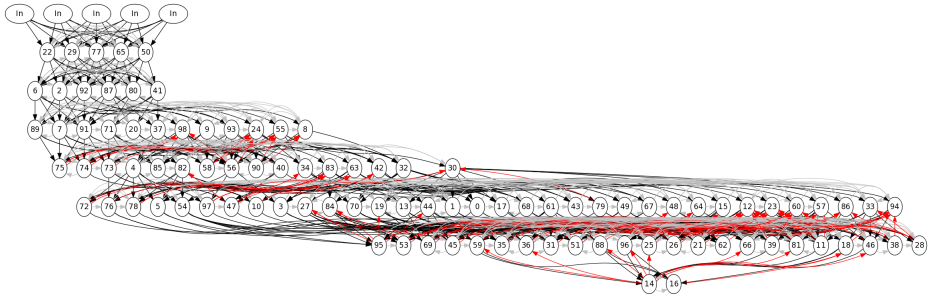
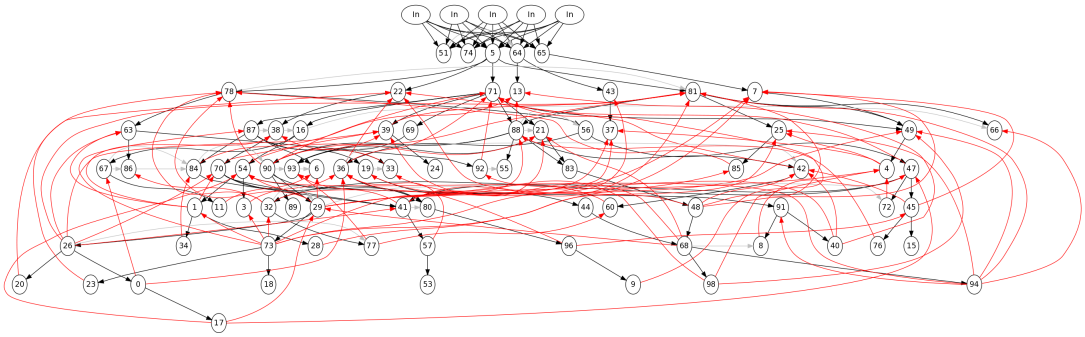


Figure 2.14: Each weight matrix has been sorted by final spike time, those that form chains display the characteristic structure as seen in previous figures. Increasing the spontaneous rate results in loopy structures.



(a) Input at 0.625 Hz Spontaneous at 0.4 Hz



(b) Input at 2.5 Hz Spontaneous at 12.8 Hz

Figure 2.15: Two example structures from simulations with the spontaneous rate higher than the input rate. Increasing the spontaneous rate increases the likelihood of loops within the structure as spike-pairs occur unrelated to the chain.

which result in reverberating activity (for example Figure 2.15). Note however that a very high input rate can also cause the development to fail. For successful chain growth the input should be between 5 and 1 Hz and the spontaneous rate less that 0.3 Hz.

2.1.3.5 Application of STDP Rule

Whether STDP should be applied multiplicatively or additively is a research topic in its own right (see Section 1.3) and can have a significant effect on the results. Here STDP is additive for potentiation and multiplicative for depression. That is potentiation steps are constant for a given spike-time difference regardless of the current weight (up to $W_{\max}$). However, depressive steps depend on the current state of the synapse; the STDP function is multiplied by weight, thus large weights fall further than small ones. This set up is thought to lead to more realistic weight distributions, specifically unimodal rather than bimodal weight distributions removing the need for hard maximum weight ($W_{\max}$ in these simulations) [160]. However, other authors have shown that when a repeating input pattern is applied, the unimodal distribution diverges into a bimodal distribution [58]. A bimodal weight distribution is also observed in the simulations here. This raises the ques-

tion, if the disparity in application between the potentiation and depression curves is not stabilising weights, does it play any other role necessary for development? Simulations were run with all four possible combinations of additive and potentiating rules.

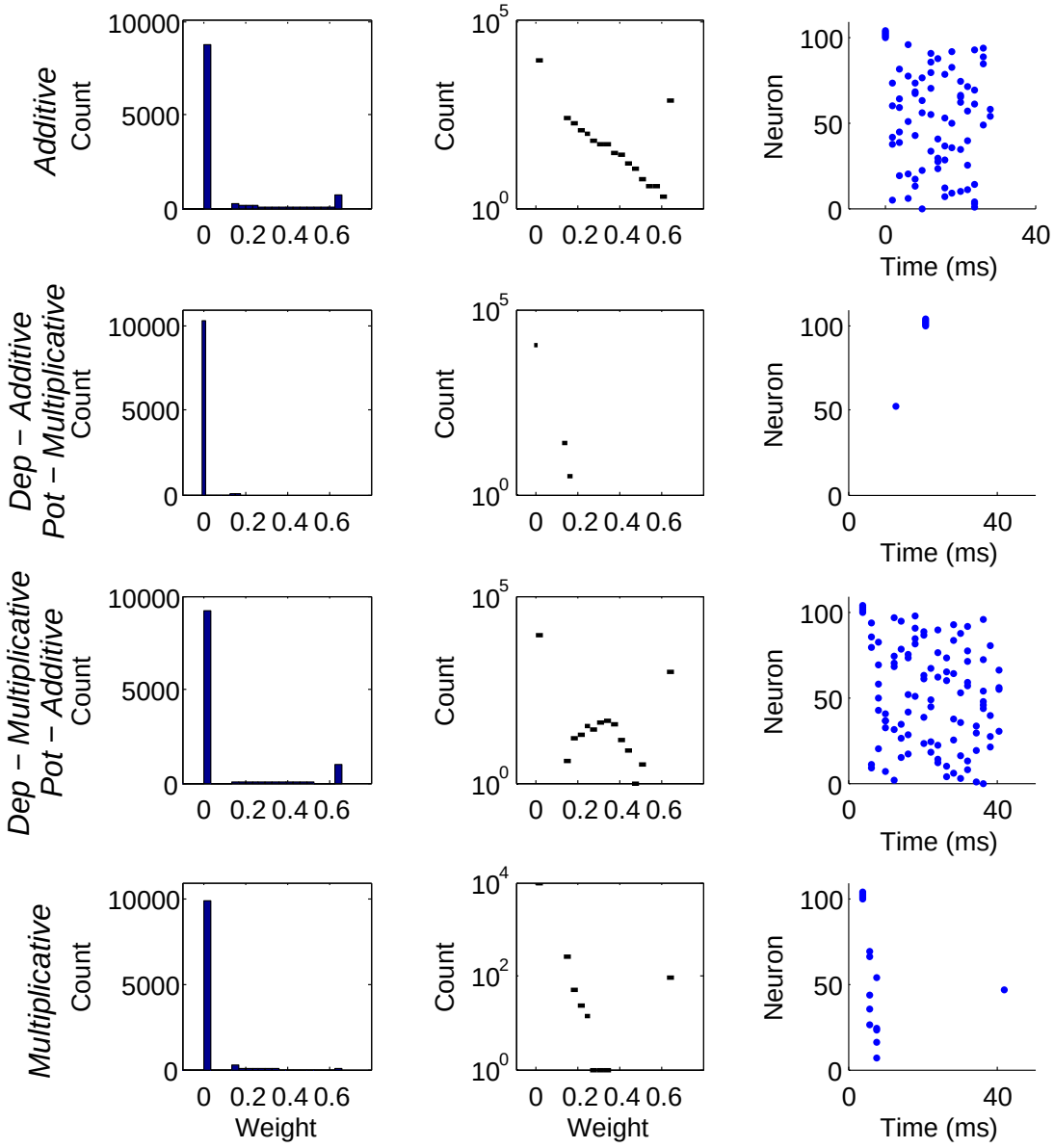


Figure 2.16: Effect of using either additive or multiplicative STDP rules. Left panels are the distribution of weights at the end of the simulation. Centre panels are the same data with count plotted on a log scale. Right panels are raster plots of final network activity.

Figure 2.16 gives the final weight distribution of simulations run with combinations of multiplicative and additive STDP. When potentiation is multiplicative and depression is additive, development does not happen at all. This can be understood intuitively, small weights can grow by only small amounts, and comparatively larger depressive steps quickly reverse any increase in weight. When both potentiation and depression are ap-

plied multiplicatively, two layers develop within the network; it is possible that given additional time further layers would develop, but longer simulations were not run. Again, the explanation is intuitive. As all synapses are initialised with low or zero weights, the first potentiation step will be small. Following this a potentiation or depression step is equally likely. If we are lucky and another potentiation follows, the weight is increased a little more. However, if this is followed by depression we will be knocked down to below the last point. Given equal number of potentiation and depressive steps, the expected net result is depression to zero. Hence only the lucky few reach the threshold for recruitment. It may be possible to counteract this effect by altering the values of τ_{LTD} or τ_{LTD} or the respective learning rates.

When either both sides are applied additively or depression is multiplicative and potentiation is additive, the simulations produce the desired chain structure. Note however the middle panels in Figure 2.16: when the weight distribution is plotted on a log scale it is possible to see that these two different approaches to applying the STDP rule give slightly different distributions in the mid-range of weights. Neither has the unimodal distribution described in [160] as clearly most of the weights are a either zero or $W_{\mathrm{max}}(0.66)$. However, disregarding the extremes, including multiplicative depression produces normally distributed weights. Those synapses that become part of an active chain are plucked from the high hand tail of this distribution and, by means of the subsequent firing induced at such weights, forced towards W_{max} and trapped there. Thus it is the act of recruitment which changes the general principle of this type of STDP rule application. In summary, for the development of synfire chains, using multiplicative depression does not appear to provide any clear advantage over the simpler additive application of both sides of the function.

A further intricacy in the way STDP rules can be applied is whether the plasticity rule is applied between all spikes or only nearest neighbour. It has been shown that in certain situations this can have significant effects of the final results [74,113]. In this model spike-time differences have been calculated for nearest neighbour only. To address whether this would affect the development of synfire chains, suites of simulations were run, identical in every way except the application of the STDP rule. The results are presented in Figure 2.17. When modelling the development of synfire chains, it does not matter whether the rule is applied between all spikes or only nearest neighbours. Due to the sparseness of firing in these models spike pairs more than one spike apart occur at such long time differences that the plasticity is negligible and overshadowed by the nearest neighbour spike pairs, thus does not alter the dynamics of development. If our model dealt with bursts of spikes (as in Ref. [38]) then the situation could be more complicated.

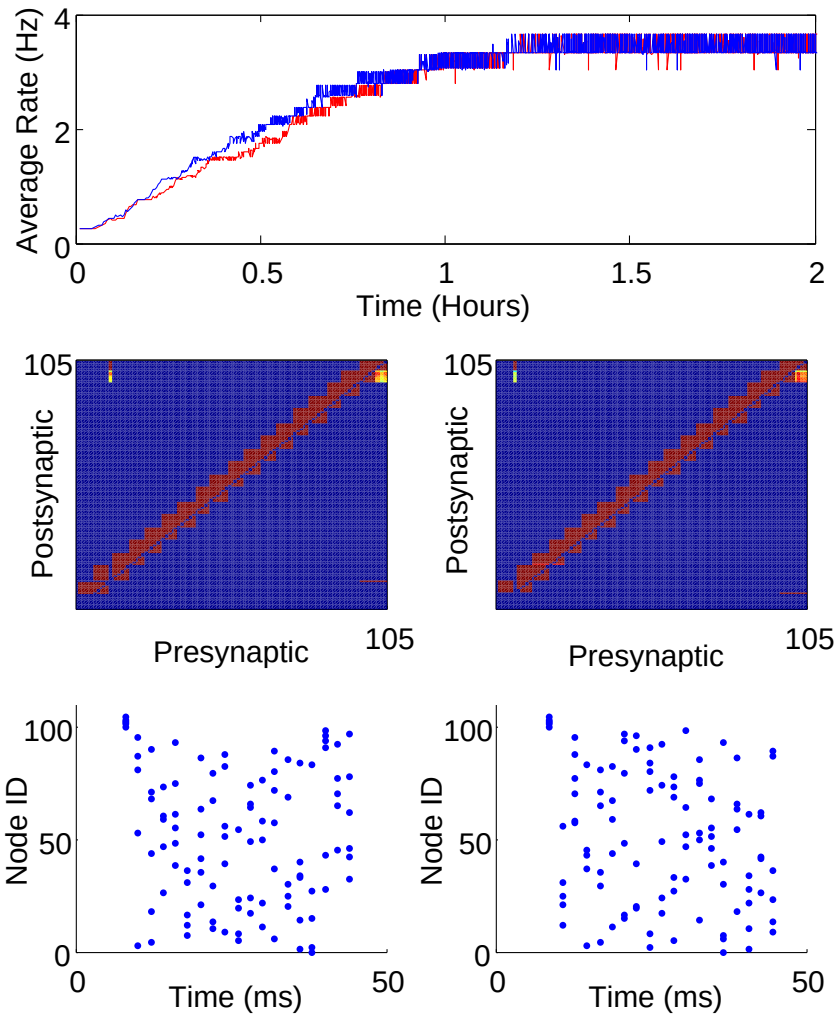


Figure 2.17: Applying the STDP at either nearest neighbour (blue in top panel and left in middle and bottom panels) or all to all (red in top panel and right in middle and bottom panels) spike pairs does not affect the simulation results. Top panel shows average firing rate, middle panels both show weights sorted by spike time and bottom panels show network activity at the end of the simulation.

Regardless of how the STDP rule has been applied in the preceding sections, the values of τ_{LTP} and τ_{LTD} have been kept at 20 ms. Although it is has become commonplace in computational modelling to use 20 ms for potentiation and depression (a practice that was begun by [143]). Actually, the range of values found experimentally differ greatly [12, 23]. It is not clear from any previous work whether the exactly matched values of τ_{LTP} and τ_{LTD}) are necessary for the development of chains, or whether they have been arbitrarily chosen. To help answer this question, simulations with a range of τ values were run. The results in Figure 2.18 show that although it is not essential for the values

to be equal, τ_{LTP} must be at least as big as τ_{LTD}. Note however, that if the ratio of τ_{LTP} to τ_{LTD} was too large, feedback connections would persist within the structure occasionally leading to loops. For chains to develop and be maintained with the network, the τ_{LTP} should be equal or slightly greater than τ_{LTD}.

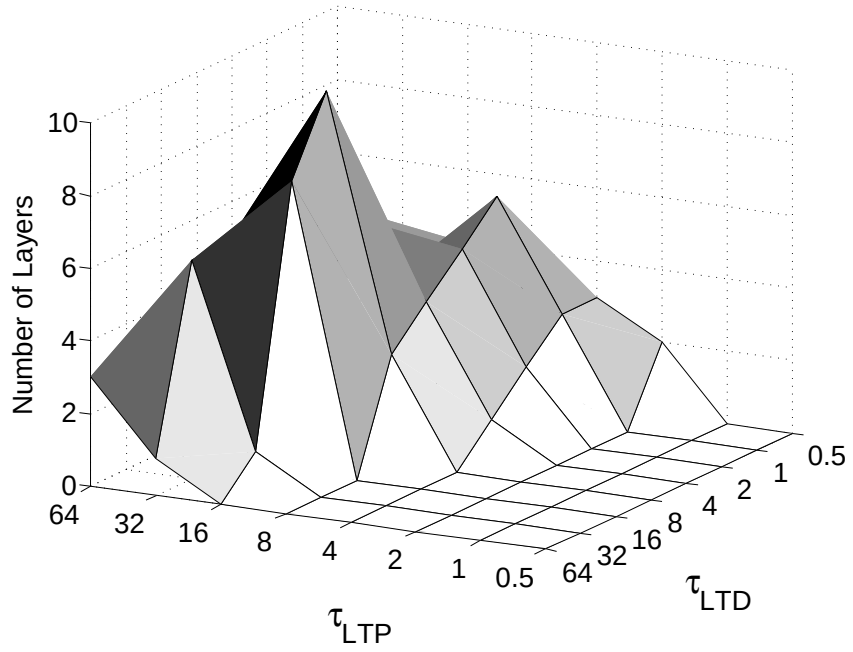


Figure 2.18: Layers only grow if the potentiation time constant is at least as big as that for depression. Peak at $\tau_{\mathrm{LTP}} = 16$ms; $\tau_{\mathrm{LTD}} = 8$ms. Results from single simulations.

2.1.3.6 Pruning

In the preceding sections, restrictions on synaptic plasticity have been imposed. Such topological constraints are common to all the successful models of synfire chain development previously discussed. In this model, the topological constraint is (as in Ref. [77]) referred to as "pruning". Here the number of synapses above a super threshold, Θ_S, is limited. In addition synapses below a further threshold, Θ_A, are effectively silent.

In all simulations with binary neurons the thresholds Θ_S and $\Theta_A$have been set at 0.133 and 0.533 respectively. To understand the importance of the values at which they are set, it is necessary to investigate the outcome of simulations using alternative values. Here the value for W_{max} was also investigated. Table 2.2 contains the outcome of these investigations. For chains to form, the super threshold, Θ_S, ust be at least 40% of the firing threshold; below this level loops are generated. In these simulations, chains only developed when the threshold below which neurons become silent was 30% of the fir-

Table 2.2: Number of layers developed in 11 hours simulated time

		g_max							
s_sup	**s_sil**	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
0.2	0.1	3	3	2	3	2	3	4	3
0.3	0.1		2	2	2	2	2	2	2
	0.2		2	2	2	2	2	2	2
0.4	0.1			4	4	4	5	4	4
	0.2			3	4	3	3	4	4
	0.3			1	1	1	1	1	1
0.5	0.1				4	5	4	4	4
	0.2				3	4	4	3	5
	0.3				1	1	1	1	1
	0.4				0	0	0	0	0
0.6	0.1					4	4	5	5
	0.2					4	4	4	3
	0.3					1	1	1	1
	0.4					0	0	0	0
	0.5					0	0	0	0
0.7	0.1						4	4	4
	0.2						3	4	4
	0.3						1	1	1
	0.4						0	0	0
	0.5						0	0	0
	0.6						0	0	0
0.8	0.1							5	6
	0.2							3	3
	0.3							1	1
	0.4							0	0
	0.5							0	0
	0.6							0	0
	0.7							0	0
0.9	0.1								5
	0.2								4
	0.3								1
	0.4								0
	0.5								0
	0.6								0
	0.7								0
	0.8								0

red text = feedback connections remain (chains will not form)

ing threshold or below regardless of the other threshold. There was a gradual increase in layer development as this threshold was reduced. Note that these simulations were run for a finite time (11 hours of simulated time) and it may be the case that the lower threshold actually increases the time it takes to develop chains rather than the capacity for development given infinite time. As the weights only become active over the said threshold then the recruitment process can only begin to occur when this threshold is reached. Hence, for a higher threshold (given the same learning rate) more coincident spikes are needed before spike transmission can happen.

The synfire chains observed in simulations so far have had layers which alternated in size between five and six nodes. As detailed earlier, this is a direct result of the limit on the number of super synapses. Thus increasing the number of super synapses allowed should increase the size of individual layers, shortening the chain for networks of the same size. Figure 2.19 shows that this is exactly the case, layer size is approximately half of the allotted number of super synapses.

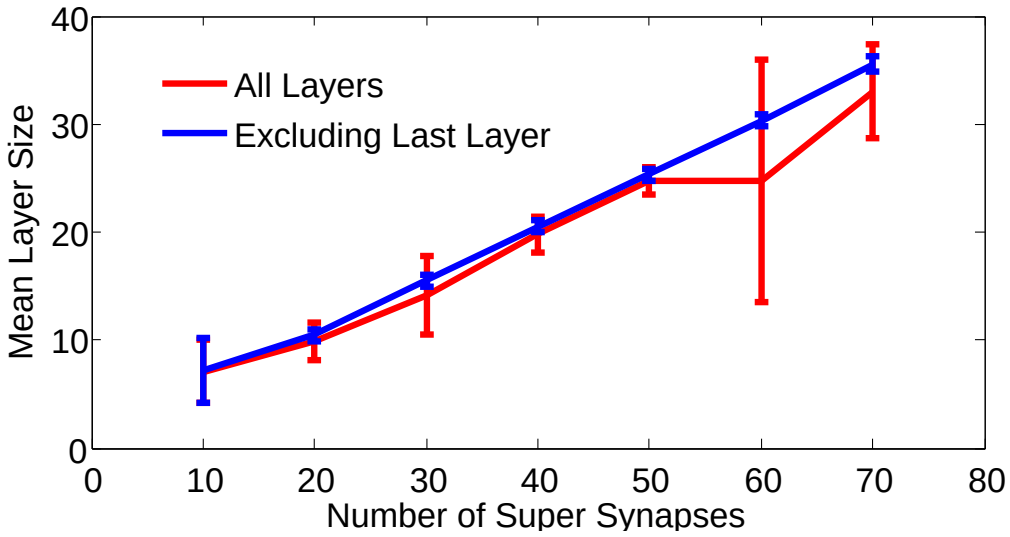


Figure 2.19: Mean layer size, error bars are standard deviation. Increasing the number of super synapses widens the size of the layers. The last layer is usually atypical as it can only grow as large as the number of remaining nodes.

If the limit on the number of super synapses is completely removed, then one large layer containing the whole network is recruited by the input. As found in [58], network development using STDP alone will result in synchronous network activity (Figure 2.20). For the development of chains with STDP, it appears that pruning or other topological constrains must also be included. Recall that, of all the models discussed in this chapter, those that grew chains with STDP also limited the total number of synaptic partners a

neuron can have, either through limiting the initial connectivity, or through topological constraints on the plasticity rule, limiting the number of total weight of synapses.

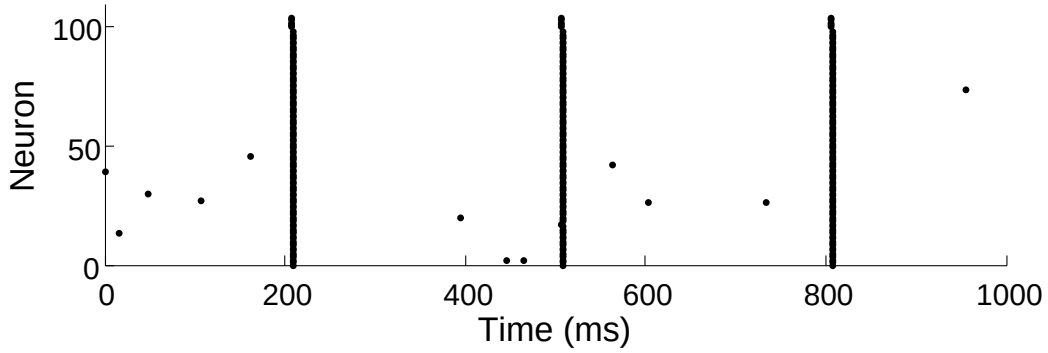


Figure 2.20: Network activity when pruning has been removed. The input recruits the entire network.

To further understand why STDP leads to the synchronisation of neural firing without topological constraints, let us consider a very small synfire chain, only three layers, each containing one neuron. The first neuron, the input, initiates the chain. With each input spike, the activity propagates down the chain with fixed propagation delays; the input is followed by the first layer, then the second. Now consider that the synapses in this fully connected network are plastic and evolve in line with the classical STDP function. With every instantiation of the input, the synapse between the input and the first layer will be potentiated, and similarly the synapse between the first and second layer. Fatally for the chain structure, as the time difference between input and layer two is positive, it too falls within the positive tail of the exponential curve and this synapse will also be potentiated. After sufficiently many repetitions of the input, the potentiation of this long range projection from the input to layer two will cause the second layer to fire directly after the input. Hence, the chain collapses (see Figure 2.21 for a schematic illustration). This small example applies generally to any size (length and width) of network, and outlines why, in networks with sufficiently dense initial connectivity, classical STDP alone is not sufficient for the development of synfire chains, or even for maintaining the stability of existing chains.

To illustrate this argument a small simulation was constructed. In this network of ten neurons, a synfire chain is already embedded. As before the network is recurrently connected with all weights set to 0, here though those synapses which form the chain, $W_{i\,i+1}$, are set to the firing threshold. The resulting spatiotemporal pattern, a string of single spikes (Figure 2.21(b)), is a result of the simple (static) chain. As the STDP rule begins to shape the network, the activity pattern changes, with increasing numbers of neurons spiking synchronously due to the reduction in the number of layers in the chain.

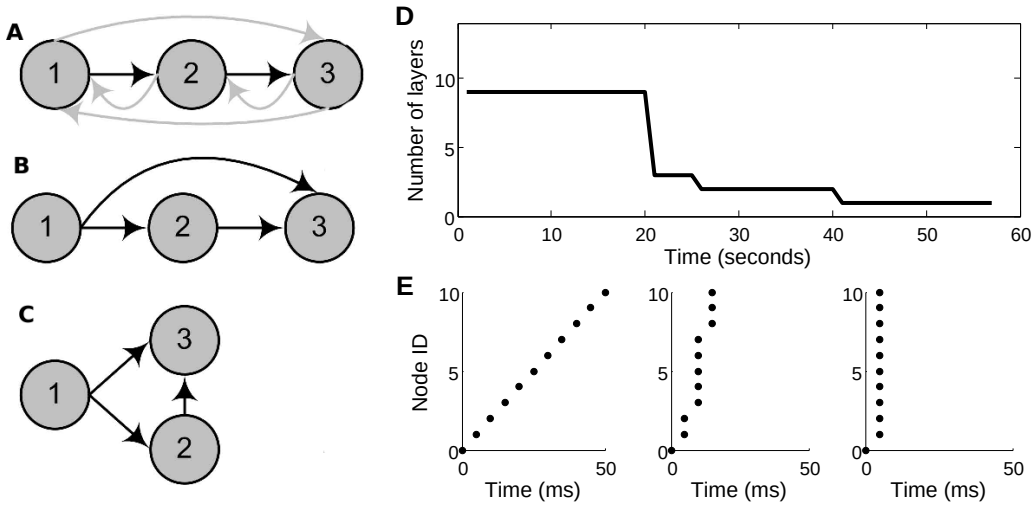


Figure 2.21: Evolution of an existing chain under classical STDP rules. For an initial network configuration (A) evolving under classical STDP, the initially subthreshold connection from neuron one to neuron three will potentiate (B). When sufficiently strong (as strong as the initial suprathreshold connection from neuron one to neuron two), spikes in neuron one will propagate in parallel to neurons two and three, causing both neurons to spike synchronously. At this point layers 2 and 3 of the chain are said to have collapsed (C). D: The number of layers in a chain structure where synapses are allowed to evolve in line with classical STDP. E: sample raster plots of the system at the beginning, after 25 seconds and after 50 seconds of simulated time.

Before long, the entire chain has collapsed: the activity pattern consists of a synchronous set of network spikes following each input spike. Pruning is necessary in this model to maintain stable chains.

It is clear that the cause of the destabilisation is the long exponential tail in the classical STDP rule. A rule without this long tail, or with a depressive regime at large positive spike-timing-differences may not lead to collapse. In the next chapter investigate an alternative STDP rule and attempt to resolve the problem of chain collapse

2.2 Conclusions

This chapter has described the development of synfire chains by presenting both the existing work in this area and new work conducted along the same lines. By creating a new model based on existing developmental models, it was possible to investigate the boundaries of models of this type helping to further understanding how various components of the model contribute towards the development of synfire chains.

It can be concluded that during development, when STDP is the main plasticity mechanism, the activity of the developing network is very important. It has been known for

sometime that developing networks display spontaneous activity [14, 37, 118, 166] and that this activity is necessary for correct development [109, 150, 154, 165]. However, here it is shown that it is not enough just for activity to be present, the relative rates of the neurons within the developing system dictate the final network structure. For synfire chains to develop, there needs to be a relatively high frequency input group and much lower frequency activity in the rest of the network. When the spontaneous activity is high interactions between network neurons create recurrent structures. Such recurrent structures, although not desired here, are known to exist within the brain (and are often used in computational models e.g., in the context of working memory [35]). Understanding how different spontaneous activity patterns contribute to generating different neural structures is an interesting avenue for future work.

Exactly how classical STDP is applied has been shown in some circumstances to have considerable effect on the results [74, 160]. Here, unlike these previous studies, the activity pattern of the developed network was structured and regular. In line with others [58] we found that result of such structured activity is a drift towards the extremes of the weight distribution regardless of how the STDP rule was applied. So in the case of synfire chain development, STDP can be calculated all to all or nearest neighbour, and either additively or additive for potentiation and multiplicative for depression. The fact that this developmental model is robust to various methods of STDP application suggests that it is a strong model with real prospects of standing up to the noise and fluctuations of real neural systems.

By step by step reduction of the existing model it has been shown that many parts of the previously published models are not strictly necessary to capture the development trajectory described. It has been shown that both the neural model and the synaptic model can be reduced and give the same qualitative results. What cannot be removed though is the competitive aspects of the development. Specifically, when using classical STDP it is essential to also include in the model, topological constraints, which limit the number of synaptic partners a neuron can have. Application of classical STDP does not lead to synfire chains, it leads to synchronised firing. This begs the question: is it possible to develop chains without topological constraints?

Chapter 3

Growing Synfire Chains without Topological Constraints: A Reduced Model

---

In the nervous system structure and function are deeply intertwined with each influencing and shaping the other. The synfire chain structure [2, 3], conjectured to be responsible for precisely timed firing sequences has attracted much recent attention [6, 7, 33, 48, 69, 72, 84, 85, 98, 139, 152, 159]. In particular, many authors have investigated how these structures might develop [13, 19, 34, 38, 58, 68, 73, 75, 77, 83, 86, 96, 105, 151]. A review of this work can be found in Chapter 1. A previously reported model of synfire chain development was explored in Chapter 2 by creating a new model including the major aspects of Ref. [77]. Briefly, a combination of intrinsic activity, classical STDP and topological constraints are shown to be necessary components of the model. It was shown that lifting the hard limit on the number of synaptic partners a neuron can have destroys the chain creating ability of classical STDP and causes networks to collapse into one synchronised layer.

Here we ask if it is possible to grow synfire chains without topological constraints. And, if so, what plasticity rules or other properties must the network employ. Having previously demonstrated that classical STDP cannot be used for this purpose, attention is turned to other experimentally observed rules. A triphasic rule is offered as an alternative to classic STDP for the production of feed-forward structures.

The chapter begins by giving the requirements for a broad range of STDP functions that offer a solution to the collapse problem. These are defined as triphasic STDP. A triphasic function inspired by experimental observations is then incorporated into a simple binary network model; here synfire chain development is observed. The generality of this model is then investigated, including work on the effect of network size, input size, learning rate and spontaneous activity. Finally the ability of the model to embed multiple distinct chains in one network is investigated.

3.1 Solving the Problem of Network Collapse: Triphasic STDP

At the end of the last chapter, the exponential tail of classical STDP was shown to cause networks to collapse into one synchronised layer unless topological constraints are also imposed to prevent such collapse. Topological constraints impose strong limits on the development and final network structure. This poses the question: Is there a plasticity rule which can grow and maintain synfire chains without resorting to restricting development through topological constraints? As the exponential tail is the cause of collapse, perhaps a rule without this long tail, or with a depressive regime at large positive spike-timing-differences may not lead to collapse.

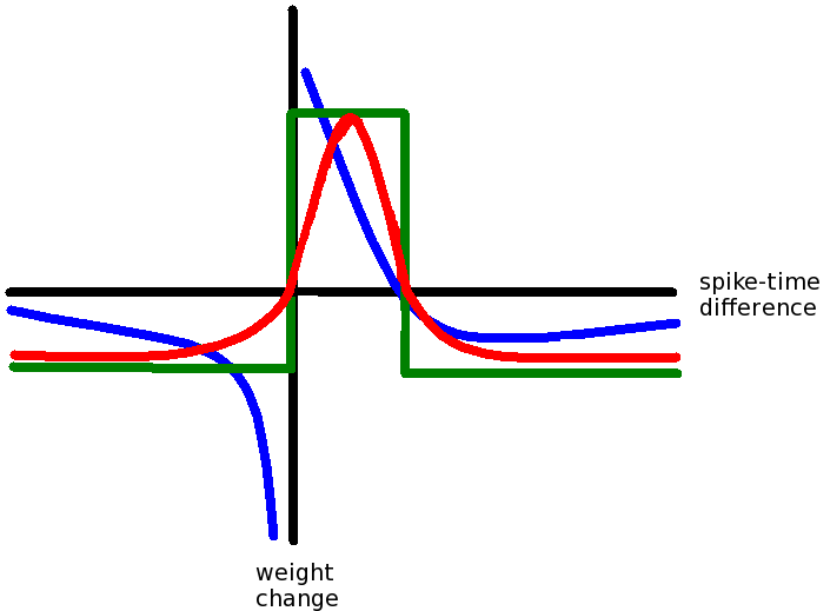


Figure 3.1: Schematic of various forms of Triphasic STDP functions.

Consider a STDP function that, like classical STDP, induces potentiation for short positive spike-time differences and depression for negative spike-time differences, but, unlike classical STDP also induces depression at positive spike-time differences that are longer than one transmission delay. One can imagine a whole host of functions that meet these criteria (See examples in Figure 3.1).

Let us first ask, would such a rule maintain exiting chains? Reconsider the small three node chain from the end of the last chapter (Figure 3.2): neuron one is strongly connected to neuron two which is strongly connected to neuron three, such that if neuron one fires, the spike is followed by a spike in neuron two and finally a spike in neuron three. The small chain is recurrently connected but the other synapses are sub-threshold. Unlike classical STDP, a triphasic STDP rule would remove all these erroneous sub-threshold connections, leaving the chain intact.

To confirm this hypothesis a small simulation was constructed. In this network of ten neurons, a synfire chain is already embedded. As before, the network is recurrently connected with all weights set to 0 except those which form the chain, $W_{i\,i+1}$, which are set to the firing threshold. The resulting spatiotemporal pattern, a string of single spikes (Figure 3.2 B), is a result of the simple (static) chain. The small simulation is run just as before (Figure 2.21). However, here the chain is maintained demonstrating that triphasic STDP (Equation (3.3) and Figure 3.7) maintains existing chains without additional topological constraints.

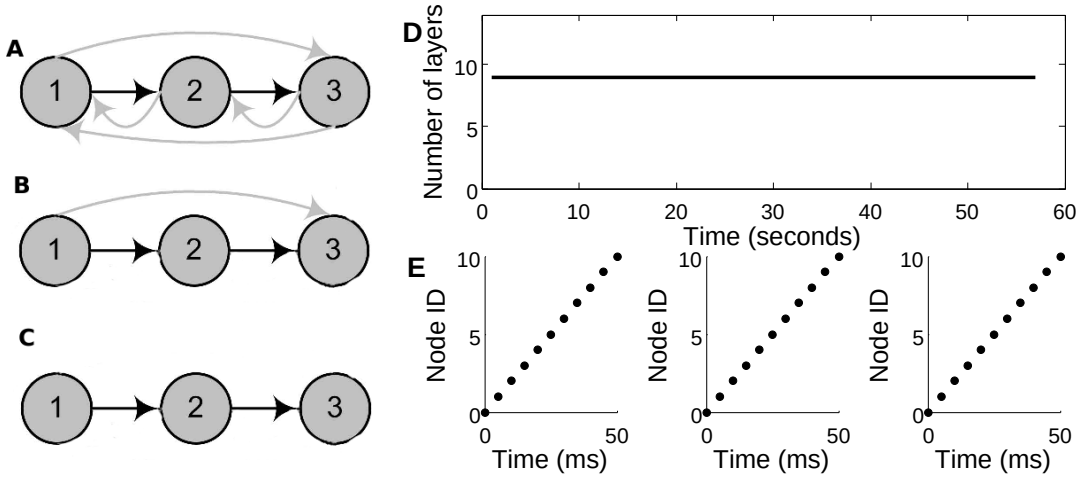


Figure 3.2: Evolution of an existing chain under triphasic STDP rules. For an initial network configuration (A) evolving under triphasic STDP, all backwards connections (B) and all long ranging forward connections will be depressed (C). D: The number of layers in a chain structure where synapses are allowed to evolve in line with triphasic STDP. E: sample raster plots of the system at the beginning, after 25 seconds and after 50 seconds of simulated time.

Table 3.1: Model Parameters

Parameter	Value
Network Size	100
Input Size	5
λ_p	0.1 Hz
λ_{in}	3.0 Hz
d	5.0 ms
θ	1
t_{ref}	6 ms
W_{max}	0.7
A	0.1
α	4 ms

This simple example shows that triphasic STDP does not disrupt existing chains in the way classical STDP does. However, the topic of this thesis is development of synfire chains. Hence a model network is constructed in which to study the development of synfire chains through triphasic STDP. The details of which follow.

3.2 The Model

The following pages contain a description of a reduced model system designed to show the possible development of synfire chains through triphasic STDP. The neural model used is very heavily reduced, ignoring the effect of time and space in dendritic integration, simply summing synchronous inputs in memoryless point neurons.

The choice of a reduced model was guided by the step by step reduction outlined in the previous chapter, the intention being to remove any variable or facet that does not directly contribute to the growth process. The details removed are done so with no implication of their relative importance in the full information processing abilities of neural systems. However, here our focus is development alone and using a reduced model allows clear and unobstructed insight into the relationship between the plasticity rule, the necessary additions to the rule and the eventual network structure.

The model was implemented in python and prototyped on a single processor of a Dell desktop PC with an Intel dual core processor. Full simulations and parameter sweeps were run on a the Leeds University HPC facility which consists of multiple Sun x84-64 based servers and storage. The code in its entirety can be found on the accompanying CD. Model parameters can be found in Table 3.1.

3.2.1 Neurons

The system consists of N pool neurons and N_{in} input neurons. The input neurons project onto the pool neurons but receive no connections. All pool neurons receive connections from the input neurons and every other pool neuron (Figure 3.3). All weights are initially set to 0, but undergo STDP (see section 3.2.2).

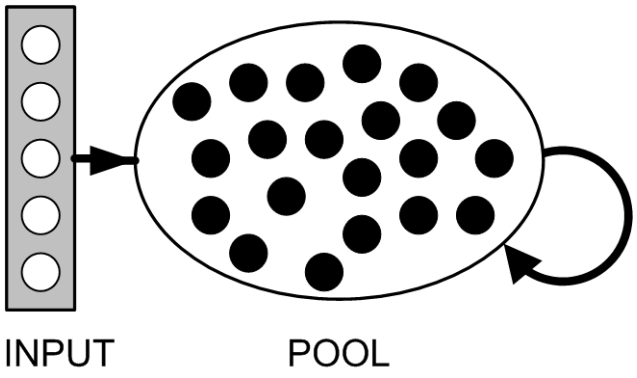


Figure 3.3: Network structure. Arrows indicate full connectivity, i.e., each pool neuron is excited by every input neuron and excites (and is excited by) every other neuron in the pool. Pool neurons fire spontaneously at a low rate. Input neurons fire synchronously and periodically at a considerably higher rate $\lambda_{\text{in}} \gg \lambda_p$.

All neurons are modelled as simple binary units, with states $S \in \{0,1\}$ and continuous-valued, instantaneous membrane potential V_j to determine neuronal spiking ($S = 1$). The potential at every point in time, t, sums over contributions from other spikes occurring precisely one time delay earlier $t - d$ and is defined as,

$$\begin{aligned} V_j(t) &= \sum_i^n S_i(t-d)W_{ij} \\ S_j &= \begin{cases} 1 & V_j \geq \theta \\ 0 & V_j < \theta, \end{cases} \end{aligned} \tag{3.1}$$

where W_{ij} is the weight of the synapse from i to j; θ is the firing threshold and d is the delay between neurons (the transmission time). Once a spike has occurred a neuron enters an absolute refractory period, t_{ref}. All synaptic weights W_{ij} are initially set to 0 and evolve according to a triphasic STDP function.

3.2.2 Triphasic STDP

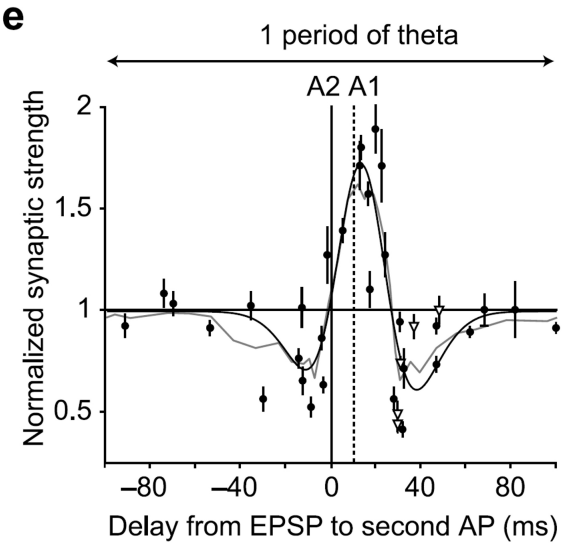
There is enormous variation between STDP rules, the form of the function varies with experimental preparation [1, 23, 128], synapse type [57, 168], neuron type [39, 156], the activity of other neurons in the local network [32] the presence of dopamine [171] and even, within the same neuron, the dendritic location [43, 93, 142]. Here inspiration is taken from an experimentally observed STDP rule which has a triphasic form [32, 116, 167]. Such triphasic STDP rule was first reported by Nishiyama and colleagues in 2000 [116]. This study was conducted on the CA1 region of rat hippocampal slices. Spike time differences of 15 ms or greater lead to depression at the synapse (Figure 3.4). Later, in 2006, Wittenberg and colleagues [167] also reported STDP with a triphasic form (Figure 3.5). Here rat hippocampal slices were also used, with CA3-CA1 synapses being investigated. When presynaptic spikes were paired with postsynaptic doublets, time differences between 25 ms and 50 ms induced depression. More recently, an even narrower potentiation window (between 0 ms and 8 ms) was reported by Delgado and colleagues [32]. In this study, pyramidal neurons in slices of primary auditory cortex where subject to simulated background conductances alongside the STDP protocol. Here a second depressive region was also observed although shallow enough to be not found significant by the authors (Figure 3.6). Interestingly, one hypothesis for the mechanism behind STDP, the calcium control hypothesis, predicts that there should be a second depressive window, which would result in a triphasic STDP rule [23, 50] (see Chapter 1).

Figures 3.4 to 3.6 illustrate experimentally observed triphasic STDP rules. Here we see a function which resembles the well known "Mexican Hat" function (the negative normalised second derivative of a Gaussian function) Figure 3.7 a) and given by

$$\Psi(t) = \frac{2}{\sqrt{3\sigma}}\pi^{\frac{1}{4}}\left(1-\frac{t^2}{\sigma^2}\right)e^{\frac{-t^2}{2\sigma^2}}. \tag{3.2}$$

Here σ is the scaling factor which determines the shape in three ways. Firstly, the amplitude or the scale parameter, determines the maximum value at t=0. Secondly, $(1-\frac{t^2}{\sigma^2})$, determines the x intercepts at σ and $-\sigma$. Thirdly, the power of the exponent, $\frac{-t^2}{2\sigma^2}$, determines the speed of return to 0.

Let us modify the function above to suit our purpose. Replacing the normalised amplitude with a arbitrary constant, A, allows us to set the learning rate in simulations independently from the width parameter. Removing the squared power from both t and σ whilst retaining a strictly positive value for the power of the exponent slows the rate of decay thus giving the broad depressive areas seen experimentally. Finally, the function is translated by replacing t by $\Delta t_{ij} - \alpha$ in all cases to obtain potentiation for small positive



e, Spike-timing dependence is explored for Δt from −100 to 100 ms. The dashed line indicates the time of the first postsynaptic action potential in most experiments. Open triangles represent high temperature controls performed at 30–34°C. The resulting learning rule has a narrow LTP window shifted in the causal direction flanked by two LTD windows. The gray curve is a moving average of four data points. The black curve is a fit to the sum of two Gaussians. Fits exclude high-temperature experiments.

Figure 3.4: Experimental observation of Triphasic STDP reproduced with permission from Ref. [167]. Note the "Mexican Hat" shaped function, shifted to the right.

spike-time differences.

Thus, we formulate a function for triphasic STDP such that ΔW, the change in synaptic weight is modified as,

$$\Delta W_{ij} = A\left[1 - \frac{(\Delta t_{ij} - \alpha)^2}{\alpha^2}\right] e^{\frac{-|\Delta t_{ij} - \alpha|}{\alpha}}, \tag{3.3}$$

where Δt_{ij} is the time difference between spikes in neuron i and neuron j. The same time delay d is used to calculate spike-to-spike transmission time and the Δt argument for the plasticity rule. α is a scaling parameter which determines the width of the potentiation window (the function crosses from potentiation to depression at 2 α) and A is the learning rate. The synapse is subject to a minimum and maximum weight of 0, and $W_{\max}$ respectively. Following the investigations in Section 2.1.3.5 we apply the STDP rule additively to nearest neighbour spikes only. For computational efficiency the STDP function is precalculated for integer values of Δt. During the simulation Δt is rounded to the nearest integer (ms) and ΔW retrieved from the look up table.

Note that here Δt is exactly the time difference between spikes and not the time difference minus the delay. Physiologically, the argument of the STDP function, Δt, is the time between the postsynaptic synaptic event (approximately equivalent to arrival of the presy-

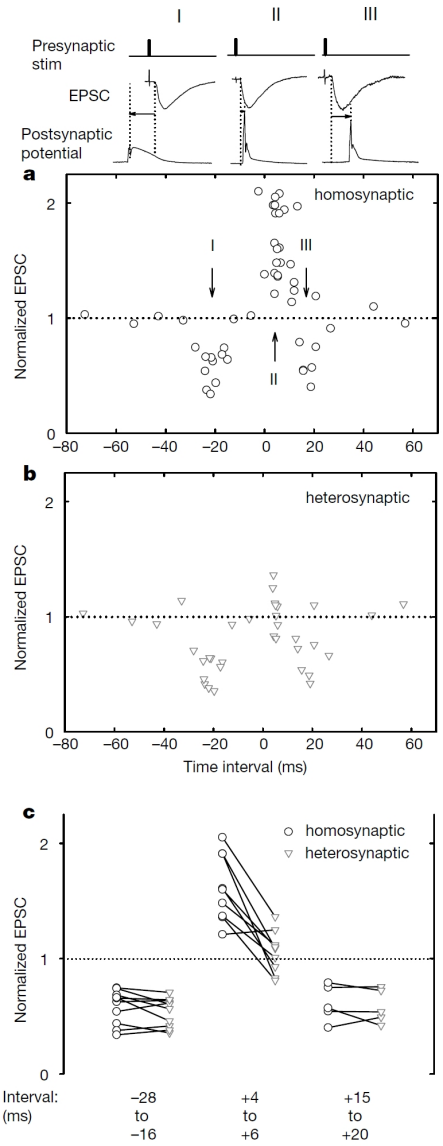


Figure 2 Critical time windows for the induction of LTP/LTD by correlated pre- and postsynaptic activation. Normalized mean EPSC amplitudes at 30–40 min after correlated activation are plotted against the interval between pre- and postsynaptic activation. The time interval refers to the time between the onset of the EPSC (top) and the peak of the postsynaptic action potential during each correlated activation. **a**, Summary of changes in the homosynaptic pathway ($n = 50$). **b**, Summary of changes in the heterosynaptic (unstimulated) pathway ($n = 34$). Data are from a subset of experiments such as those shown in **a**, in which the control pathway was monitored throughout the experiment. **c**, Changes of synaptic strength at the homo- and heterosynaptic pathways for experiments using time intervals of –28 to –16, +4 to +6 and +15 to +20 ms. Lines connect data points from the same experiment.

Figure 3.5: Experimental observation of Triphasic STDP from Ref. [116]. Note the "Mexican Hat" shaped function in panels a and b and the second depressive region clearly summarised in panel c. Reprinted by permisson from Macmillan Publishers Ltd: Nature [116], copyright (2000)

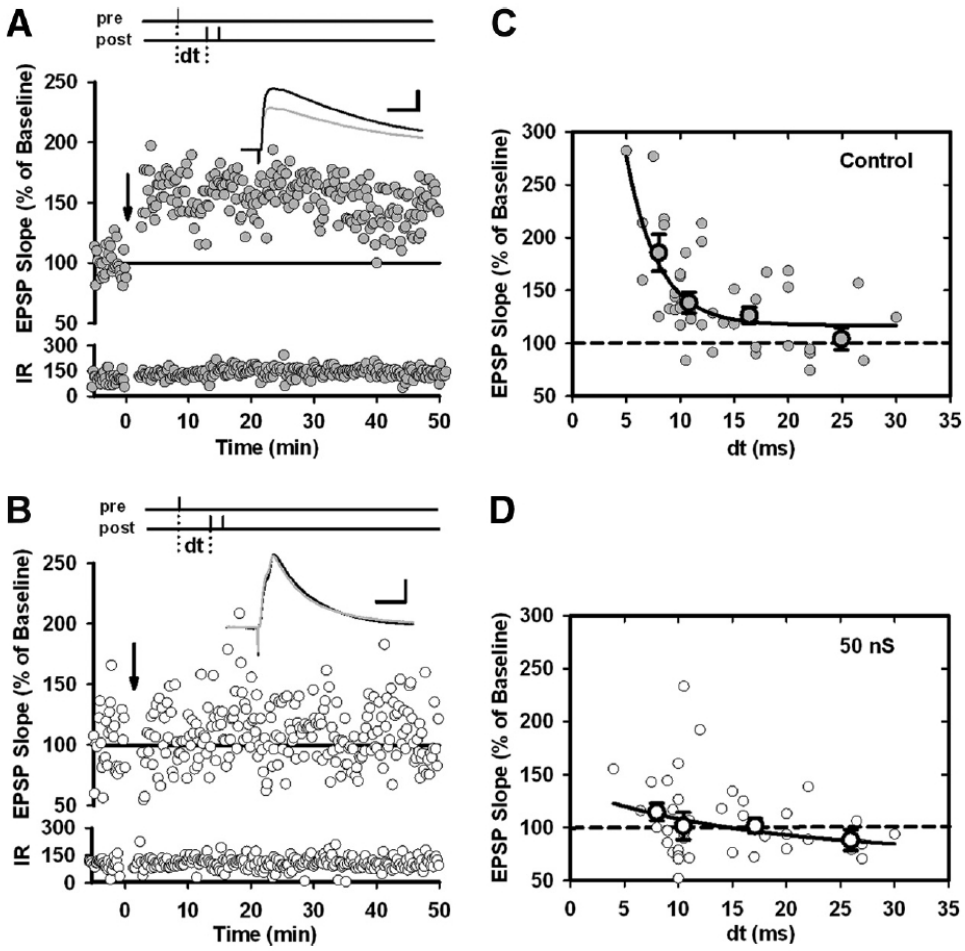


Figure 2. Membrane conductance sharpens the potentiation side of the STDP curve. ***A***, Single representative experiment showing the change in synaptic strength after the induction of t-LTP in a control cell. t-LTP was induced after a stable 5 min baseline by pairing EPSPs and bAPs with a delay of +10 ms (*dt*). The arrow indicates pairing onset (pairing frequency, 1 Hz; 100 total pairings). Traces show the synaptic responses evoked before pairing (gray) and after pairing (black). At the bottom is the input resistance (IR) (in megaohms) over time. ***B***, Single representative experiment showing the change in synaptic strength after the induction of t-LTP in a cell injected with 50 nS shunting conductance. ***C***, Change in EPSP slope for individual t-LTP experiments performed on control cells as in ***A*** but with various pairing delays (*dt*; $n = 44$). Also shown are mean changes for experiments grouped into four ranges of delays: 5–9.5 ms, 10–12 ms, 13–20 ms, and >20 ms. ***D***, Same as ***C*** but for t-LTP experiments performed on cells injected with 50 nS shunting conductance ($n = 39$). Calibration: horizontal, 20 ms; vertical, 5 mV. Error bars indicate mean ± SEM.

Figure 3.6: Experimental observation of Triphasic STDP reproduced with permission from Ref. [32]. Note the depressive region for positive spike times in panel D.

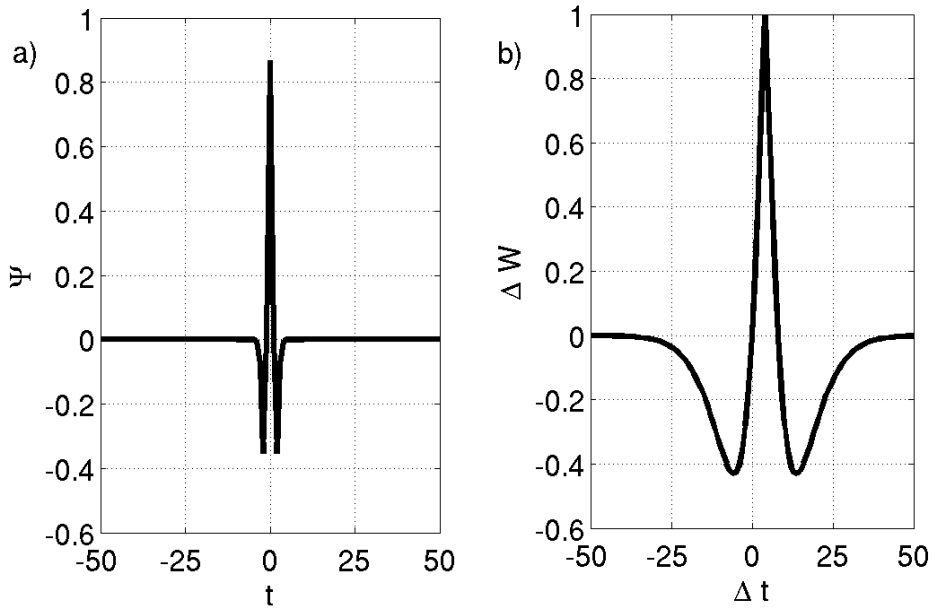


Figure 3.7: a) The Mexican Hat Function with σ at 1. b) Triphasic STDP with α set to 4 ms and A set to 1.

naptic spike at the synapse) and the arrival of the back propagating action potential at the synapse. Using the spike-time difference minus the delay (as in Chapter 2) represents the delay as entirely axonal and will result in a spike-time difference of 0 ms for postsynaptic spikes generated exactly one delay after a presynaptic spike. Hence neglecting the physiological time. Here, it is assumed that synfire chains are embedded in small microcircuits, such that the axonal delay, the dendritic delay and the time course of back propagation are all of equal duration. The axonal delay plus the dendritic delay is therefore approximated as equal to the dendritic delay plus back propagation [114]. Hence spike-time difference is used as a proxy for the STDP argument.

3.2.3 Spontaneous Network Activity

A commonly observed feature of developing neural tissue is spontaneous activity [14, 37, 118, 166]. Such activity which is present even before synapses become active, is characterised by highly non-stationary spiking activity, with brief network bursts between longer periods of relative quiet. It is thought that this early intrinsic activity has an important role in guiding the development of neural tissue and it has been shown that removal of such activity disrupts the normal developmental process [109, 150, 154, 165]. Clearly, if development is to be driven by STDP, there must be spiking activity within the immature

system. In the model presented here spontaneous activity is included within the network by forcing pool neurons to fire with to a Poisson spike-time distribution with rate λ_p (up to the neurons' refractory period t_{ref}, with $\lambda_p r \ll 1$).

Experimental observation has also shown that spontaneous activity, whilst critical to development, diminishes once neurons are part of an active network [109, 150, 154, 165]. This cessation of spontaneous activity is included in the model and is referred to as "activity dependent excitability". Practically, this means that once a neuron is part of a chain (i.e., the neuron fires as a result of presynaptic activity) the spontaneous activity is turned off.

In contrast to pool neurons, input neurons fire regularly with rate λ_{in}. The activity of these neurons is fixed and does not change throughout the simulation.

3.3 Results: Synfire Chain Development

Using the above model it is now possible to test the effect of triphasic STDP within a simulated network. A network is constructed as outlined above, All other parameters are as in Table 3.1. Observing the initial network activity reveals the regular firing of the inputs and the sparse firing of the pool neurons (see Figure 3.8). As all of the synapses at this stage are silent (have a weight of 0) only that activity which is intrinsic to each neuron type is observed.

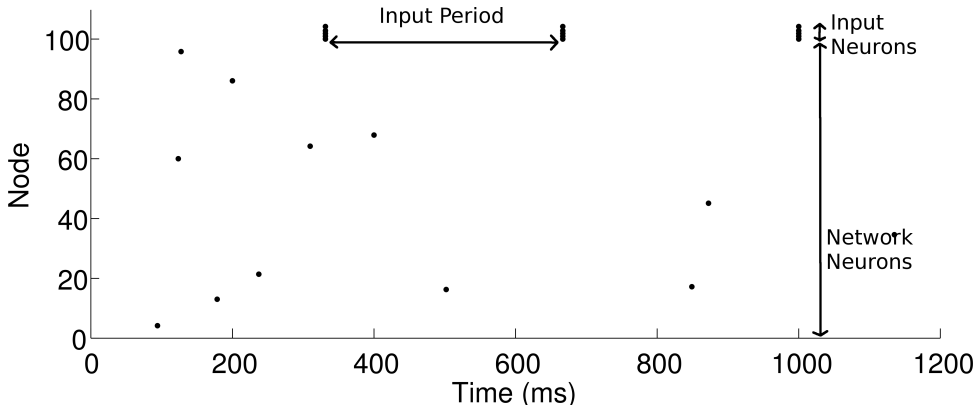


Figure 3.8: A raster plot of initial network activity

Development of the first layer of a synfire chain through recruitment of network neurons by the input is a very similar process to that which occurs under classical STDP. If a spontaneous spike occurs at one of the pool neurons directly ($< 2\alpha$ ms) after a spike in one of the input neurons, then the synapse from the input to the pool neuron is potentiated. With triphasic STDP there is a much larger range of spike time differences that lead

to depression. Hence, it is generally the case that the potentiated synapses will depress again to zero. However, if several potentiation steps occur with relatively few intervening depressive steps such that the incoming synapses onto the pool neuron sum to a combined weight that is sufficient to induce a spike ($\sum W \geq \theta$), then the pool neuron fires one delay, d, after the input. At this point the neuron is "recruited" to the first layer of the chain and will remain in this position. If d and α are set so that the resulting spike is within the potentiation window of the STDP function then each further presentation of the input will result in a spike in the pool neuron d milliseconds later and further potentiation (up to $W_{\max}$). Unlike models which rely on topological constraints to limit the synaptic activity of recruited neurons, here all synapses onto or from the neuron will continue to develop according to triphasic STDP throughout the simulation.

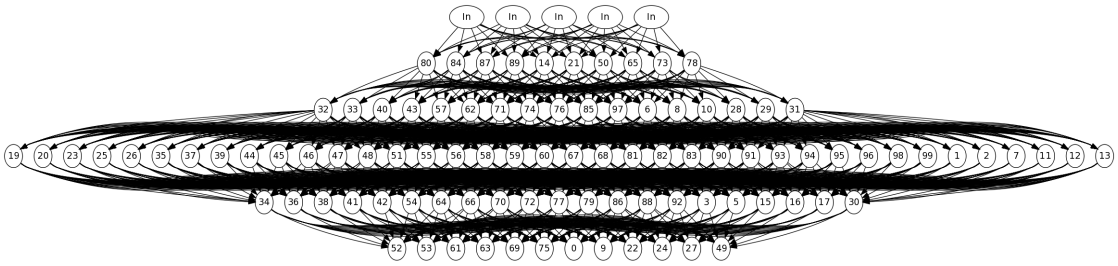


Figure 3.9: Network diagram of a completely developed network all synapses with weight above 0.53 are drawn. Neurons are labelled with the ID assigned at the beginning of the simulation. The network develops into an entirely feed-forward structure.

Due to the fixed delay and the deterministic neural model, once recruited to a particular layer, the neuron's spike time relative to the input does not change. The neuron will fire at this relative time for all future presentations of the input. Neurons recruited into the first layer now spike at the same rate as the input. Therefore, the process described above can now occur between either the input and an unrecruited pool neuron or between the first layer and an unrecruited neuron. This process takes place in tandem at each additional layer until the network is fully formed (Figure 3.9).

As the chain grows, each layer is able to recruit any of the unrecruited pool neurons which remain. Every layer consists of perfectly synchronised neurons (Figure 3.10), resulting in identical potentiation or depression events at the synapses between each member of the layer and any particular pool neuron. As a result, the effect of each plasticity step on the downstream neuron is multiplied by previous layer size. Because of this, the synapses from layers with few neurons must reach higher weights to induce spikes in (and consequently recruit) pool neurons. Therefore, recruitment onto the second layer will accelerate as the first layer grows, and so on for further layers. Note that with no hard limit

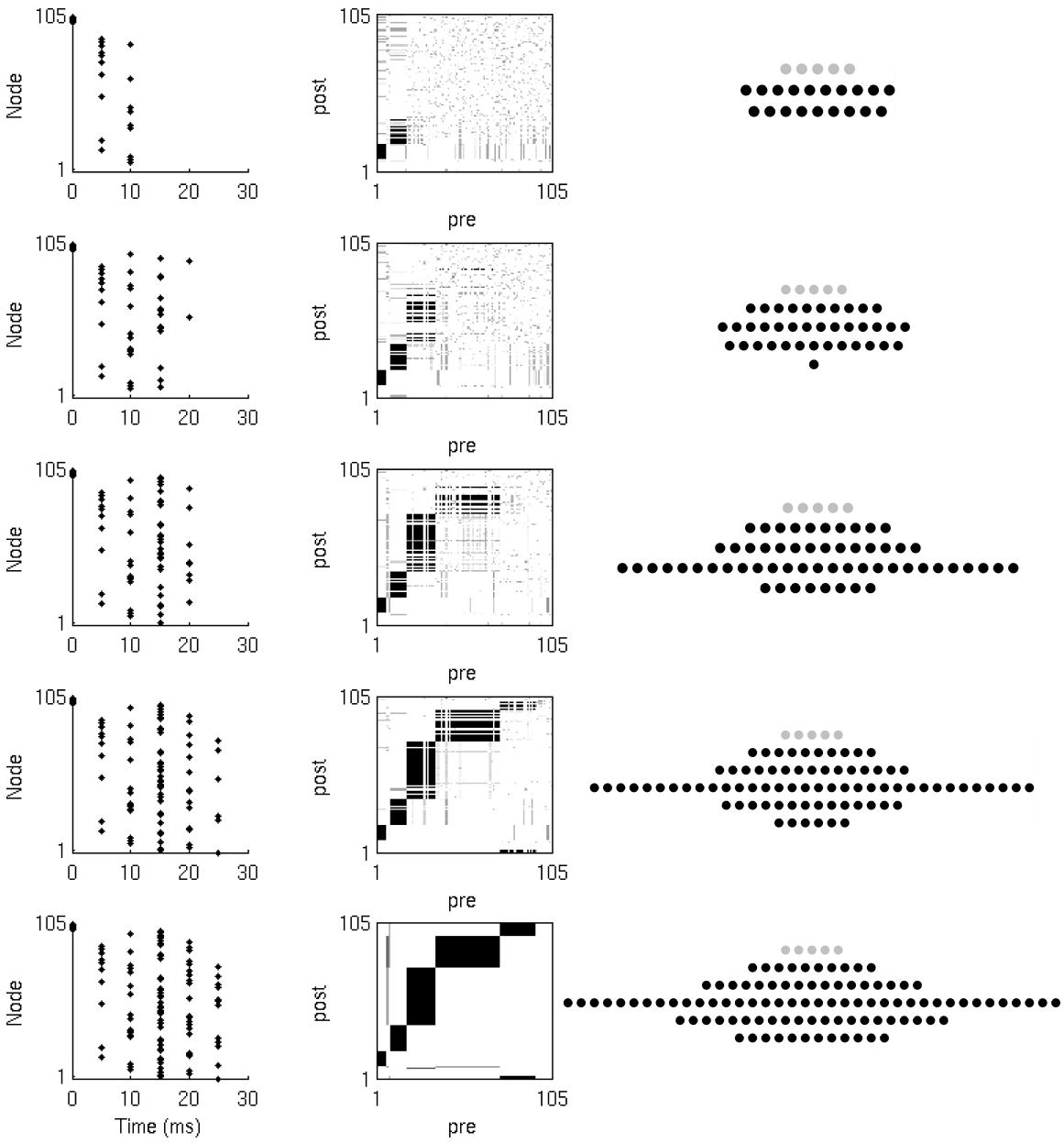


Figure 3.10: Network activity and synaptic weights. Network activity (left), weight matrix evolution (centre) and network structure (right) during synfire chain development (from top to bottom). Neuron indices in weight matrices have been sorted by spike time order (in response to the last input in the simulation), and thus represent the neuron's position in the developed chain. For clarity connections betwen neurons have been removed from the network structures (right). However, they are feedforward as in Figure 3.9 and centre.

on the number of neurons in a layer or on the number or strength of the efferent or afferent synapses of a neuron, recruitment continues to all layers throughout the simulation. In other words, layers are added to the end of the network, whilst at the same time, existing

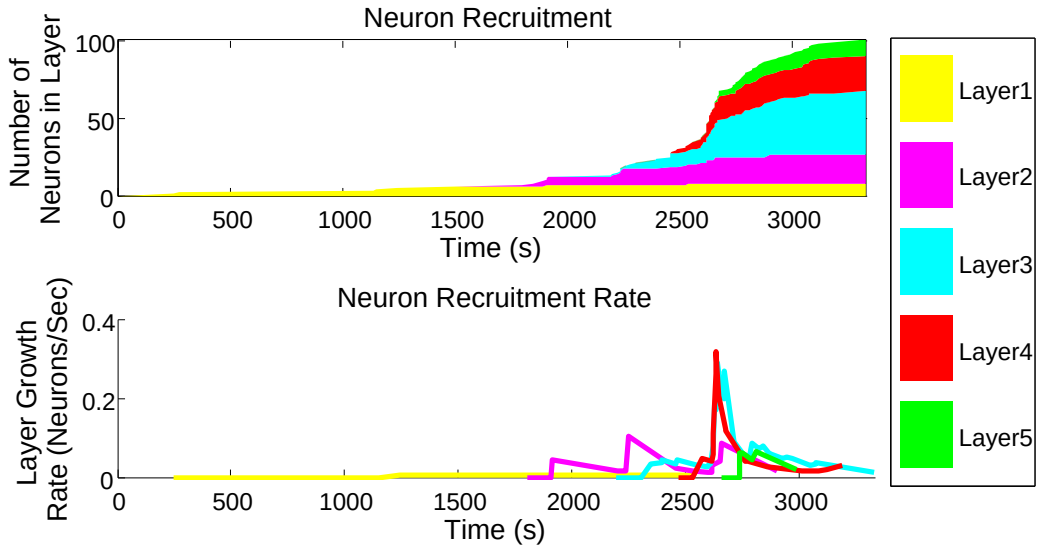


Figure 3.11: Neuron recruitment by layer. Top: each coloured wedge gives the size of the layer through time. Bottom: the rate of recruitment depends on the size of the previous layer at that time and available pool size. Recruitment rate calculated per second by rolling average from the previous four recruitment events

layers continue to grow in size, until all neurons are recruited onto the chain.

Importantly, the rate of recruitment at each layer depends on the size of the previous layer (Figure 3.11). In this example layer 3 (blue) becomes the largest, inducing fast growth in layer four (red). Eventually, all neurons are depleted from the pool (onto the maturing chain). The progressive modulation of each layer‘s recruitment rate leads to a characteristic chain structure, as shown in Figure 3.16 for a variety of network sizes. this is discussed further in Chapter 5.

3.3.1 Comparison of Triphasic STDP and Classical STDP

The are many similarities between the above description of synfire chain development with triphasic STDP and the development of synfire chains through classical STDP. In the previous chapter and in refs [13, 38, 68, 77] connections between an input group and the undeveloped network grow through the culmination of random spiking events. Unlike classical STDP, here further layers are added to the chain through the properties of the STDP rule alone. Stable further layers requires additional topological constraints when using classical STDP.

Figure 3.12 gives the weight trajectories for the synapses from a single input node to the entire network. Here each synapse rises and falls in line with spiking events encoun-

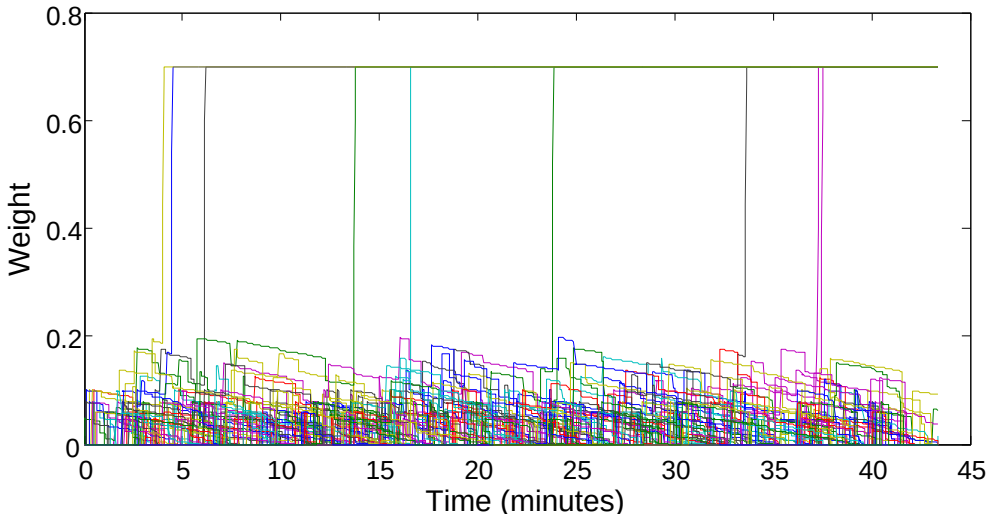


Figure 3.12: The synaptic weights from an input node onto the network, each line tracks the weight trajectory over time. Note how, in contrast to Figure 2.6, the weights hop up and down during development. Data from a single run with parameters as Table 3.1.

tered, and appears to approximate a variable step, biased and bounded random walk. If a synapse reaches an upper bound (0.2), where an input spike will cause a postsynaptic neuron to spike, it is quickly potentiated to the maximum value. This trajectory is very different to that displayed in Figure 2.6 where, under classical STDP, all of the synaptic weights are increasing over time. There, it is the topological constraints that introduce competition between neurons; here it is the STDP rule itself.

In previous work (e.g. [38, 77]) the network structure was determined by the topological constraints. specifically, the number of neurons in a layer was dependent on the cap on synaptic partners. In this new model, these hard constrains have been removed and the resulting chain structure is determined by the STDP rule and the number of available neurons. This is therefore the first example of a synfire chain development model in which the network structure is determined by the STDP rule and not the topological constraints.

Over the next few sections the details specific to this system are explored.

3.3.2 Interaction Between Delay and Triphasic Potentiation Window Boundary

Critical to the success of growing synfire chains through triphasic STDP is the interplay between the crossing point from potentiation to depression for positive time differences,

2α, (see Figure 3.7) and the time delay, d, associated with spike propagation between two neurons. Specifically, single delays must result in potentiation while multiple delays must depress, such that $\alpha < d < 2\alpha$.

This interdependence between the delay and the STDP function can be seen experimentally, as shown in Figure 3.13. In simulation, when the delay is set to less than α, then far forward projections (i.e., those that traverse more than one layer) are also potentiated, causing a collapse to a single layer, a similar process to the collapse seen when using classic (exponential) STDP. Additionally, when the delay is set to greater than 2α then the single layer forward projections are always depressed, such that no layers form at all. Transmission delays in the range α to 2α will result in chain growth with no significant variation in results between values within this range. (Here transmission delays are always uniform for any particular network. See Section 4.2.2.2 for networks with heterogeneous delays)

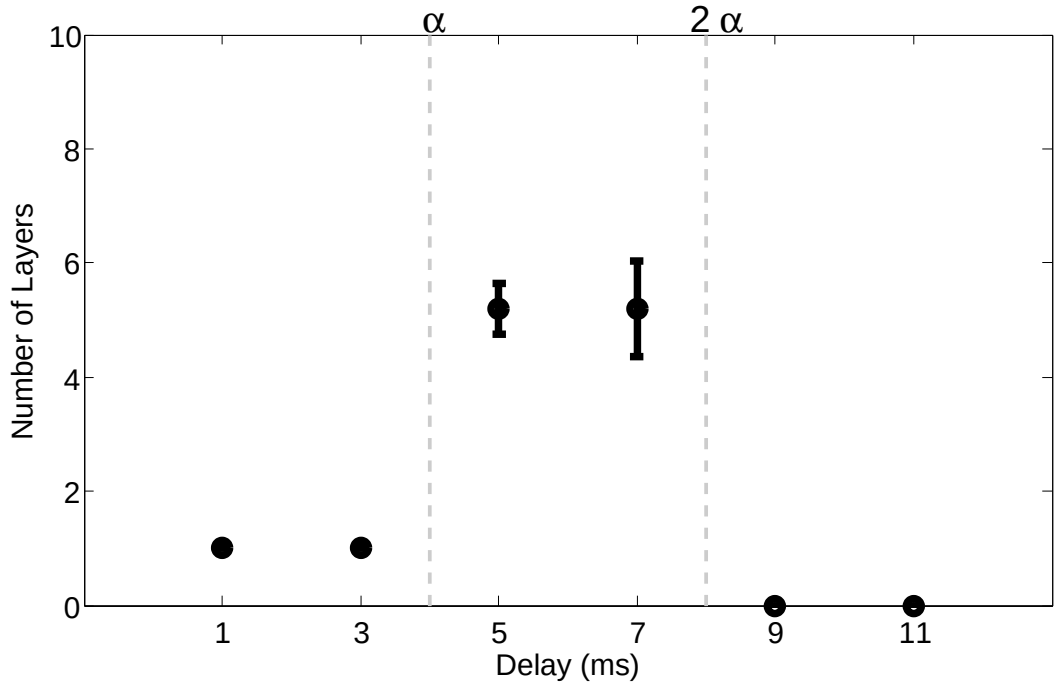


Figure 3.13: Transmission delay and network growth. Simulation parameters as Table 3.1 except d, which varys. When α is set at 4 ms, chains only grow when the transmission delay is between 4 ms and 8 ms. Error bars are standard deviation on 5 runs.

3.3.3 Network Size

So far the results presented have been obtained from simulations of 100 neurons only. Clearly microcircuits within the brain are be much larger than this. Therefore, chain building through triphasic STDP should be robust to scaling of the network size.

3.3.3.1 Looping

As the network size is increased, there is an increasing chance that feed-back connections, created randomly, will remain. For example, a backwards connection that traverses many layers will not be removed as the depressive regions tend to zero after ± 50 ms. As the synapses between pool neurons fluctuate according to their Poisson spike times, it is possible that, on recruitment, a neuron has connections to other parts of the network (either backwards or far forward). In small networks (less than 800 neurons), this is not an issue, because these connections fall within the depressive window of the STDP rule. However, in larger networks these connections are a problem for two reasons. Firstly, unless other parameters (input size, learning rate etc.) are adjusted as network sizes grow there will be more layers in larger networks, so the possible spike time difference from input to the last layer can be much larger thus resulting in negligible depression due to STDP. Secondly, the width of the layers is greater in larger networks, so small effects (due to a single spontaneous spike) can be magnified. When a neuron is recruited that has positive connections onto an existing layer, even if the weights of the individual synapses are small, if there are many of them (because the layers are wide) then this can cause spiking within recruited nodes, but in the incorrect place. Either of these scenarios can result in a loop within the chain. A loop results from a neuron projecting backwards to another neuron that resides in a earlier layer. If a loop is maintained the network enters a pathological state that results in continuous and repeating network activity (Figure 3.14). As triphasic STDP is applied at all times, and not limited by topological constraints, a change in the activity pattern will result in a change in the network structure. Repeating activity will break down the long chain and create a recurrent structure.

In the brain, synapses that are not active are known to recede [100, 106, 148]. Computationally, this can be modelled in a number of ways. An activity dependent plasticity function [96] can be implemented that depresses those synapses that are not active. Homeosynaptic depression [77] can be implemented in which all synapses are depressed at a very low rate. In this case, those that are active (thus undergoing frequent potentiation) will compensate for this depression; those that are quiet will slowly become silent.

In this model, for computational simplicity, we incorporate the silencing of non-active synapses into the STDP rule. To do this we assume that all very large spike-time differences $|\Delta t| > 50$ ms are uncorrelated and those synapses are depressed at a low level. Synaptic plasticity for such large (positive or negative) spike-time differences is uniform and equivalent to that a ± 50 ms respectively. Thus those synapses potentiated through infrequent spiking are depressed over time (Figure 3.15). The Triphasic STDP function

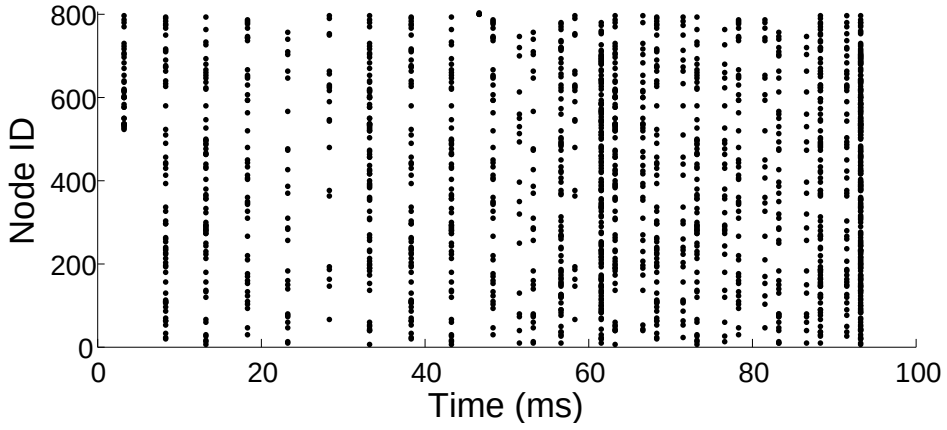


Figure 3.14: Network of 800 pool neurons with 5 input neurons. The network has developed recurrent connections and its activity becomes self sustaining.

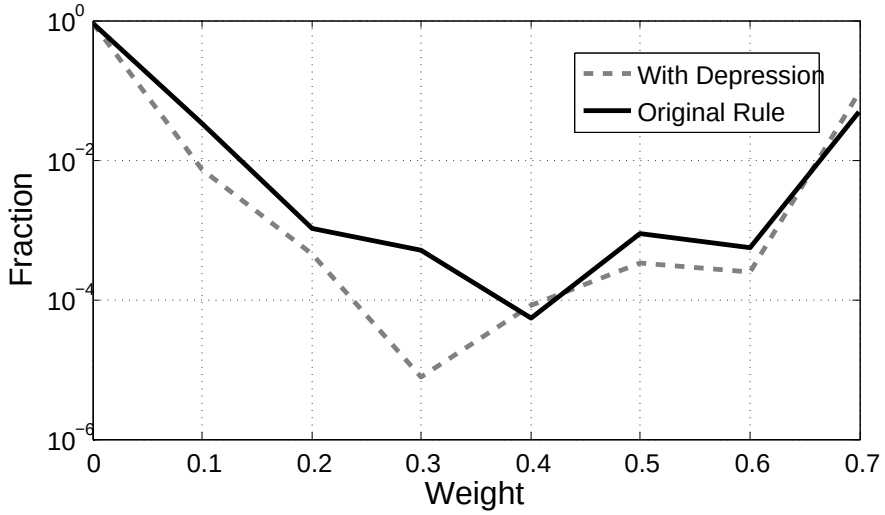


Figure 3.15: Adding depression for long spike-time differences to the STDP function reduced the proportion of weights in the mid-range removing the possibility of loops forming. (10 runs, 1000 neurons)

is henceforth defined as,

$$\Delta W_{ij} = \begin{cases} A\left[1 - \frac{(-50-\alpha)^2}{\alpha^2}\right] e^{\frac{-|-50-\alpha|}{\alpha}} & \Delta t_{ij} < -50\text{ms} \\ A\left[1 - \frac{(\Delta t_{ij}-\alpha)^2}{\alpha^2}\right] e^{\frac{-|\Delta t_{ij}-\alpha|}{\alpha}} & -50\text{ms} \geq \Delta \text{t}_{\text{ij}} \leq 50\text{ms} \\ A\left[1 - \frac{(50-\alpha)^2}{\alpha^2}\right] e^{\frac{-|50-\alpha|}{\alpha}} & \Delta t_{ij} > 50\text{ms}. \end{cases} \tag{3.4}$$

3.3.3.2 Scaling

With the above addition, we can now examine the consequence of increasing network size. In Figure 3.16 the final shape of networks containing between 250 and 1000 neurons are compared. To ensure fair comparison, the number of input neurons was scaled with the number of pool neurons. In each case the ratio of input to pool neurons was 1:50. Given the much increased number of input nodes, to ensure recruitment continued at the same rate, it was necessary to reduce the learning rate by the same factor. Doubling the number of pool neurons doubles the number of input neurons and halves the learning rate.

Figure 3.16 suggests the networks produced by this model are scale-invariant. The network size can be increased without affecting the relative size of the constituent layers. Increasing the size of the network, given the appropriate scaling of input and learning rate as described, increases the size of each layer, but does not affect the number of layers or the overall network shape. It is predicted that networks could be scaled to arbitrarily large sizes and the chain structure would remain the same, with relative layer sizes as observed here, possibly with the addition of some further small layers at the tail end. Clearly, in biological networks, this would be a problem. One does not expect to see the entire brain recruited to one single chain. However, here only the mechanisms necessary to grow chains are modelled. In the brain, there are many competing mechanisms that would act to restrict chain growth. Competing inputs is an example considered here (Section 3.3.7) but there are many more including inhibition, connectivity restrictions and genetic factors which would make interesting questions for further work.

To understand why the input size and learning rate must be similarly scaled, it is important to understand their contribution to the final network shape.

3.3.4 Input Size

As illustrated in Figure 3.17 the shape of the final network can be modified by adjusting the proportion of input neurons to pool neurons. With fewer input neurons the resulting chain contains more layers. The distribution of layer size is also altered, with fewer inputs

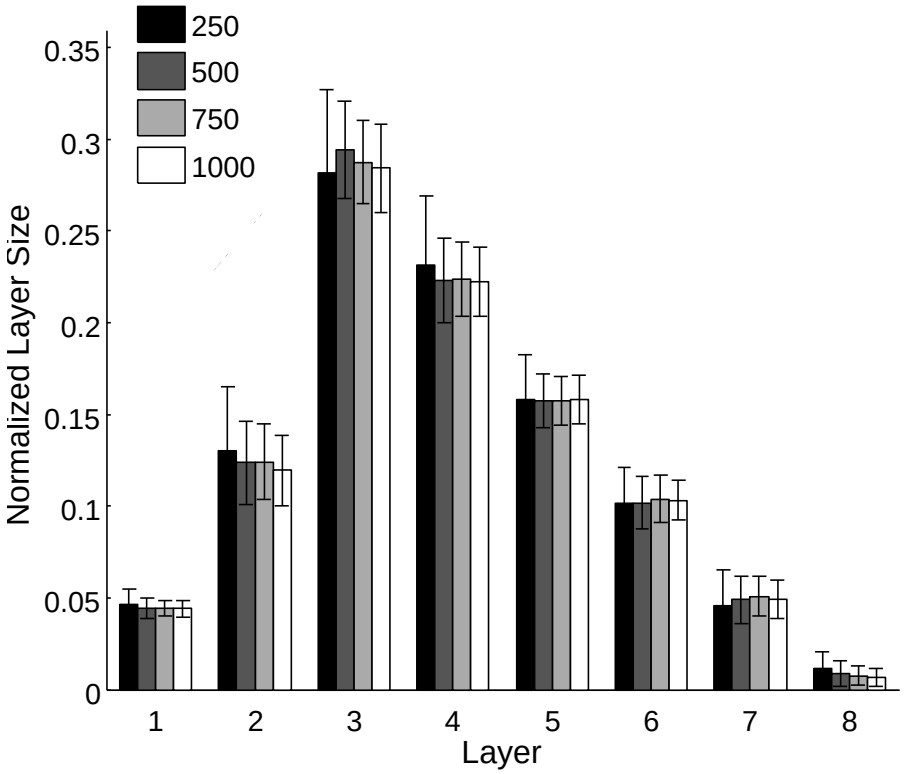


Figure 3.16: Scaling of Network Size. Mean layer sizes $\pm$ standard deviations for different size networks (100 runs). Layer sizes are normalised by the number of pool neurons (see legend). The number of input neurons and learning rate are scaled accordingly. The chain structure follows a consistent pattern that scales with network size, preserving the number of layers and their relative sizes.

the largest layer is located further from the input. With more inputs the largest layer is closer to the input.

The modulation of network structure through input size is a result of the 'race to recruitment'. I.e., as each neuron within the pool can be recruited to at most one layer, there is competition between the layers to recruit neurons and hence increase the size of the following layer. As each spike of a pool neuron results in multiple identical plasticity events between that pool neuron and all members of an existing layer, the effect of any downstream spike will be the potentiation step (given by Equation (3.4) and determined by the spike-time difference) multiplied by the number of neurons in that downstream layer. Hence, those layers with many neurons require less such events before recruitment occurs: They have a faster recruitment rate.

The result of the above process is that when the input is small, the first layer becomes

larger than the input layer relatively early on in the simulation, shortly followed by second and so on. The consequence is that in the final network configuration, the largest layer is some distance from the head of the chain. Conversely, when the input is large, many neurons must be recruited before the "race leader" shifts downstream. Thus the resulting network has its largest layer relatively close to the input.

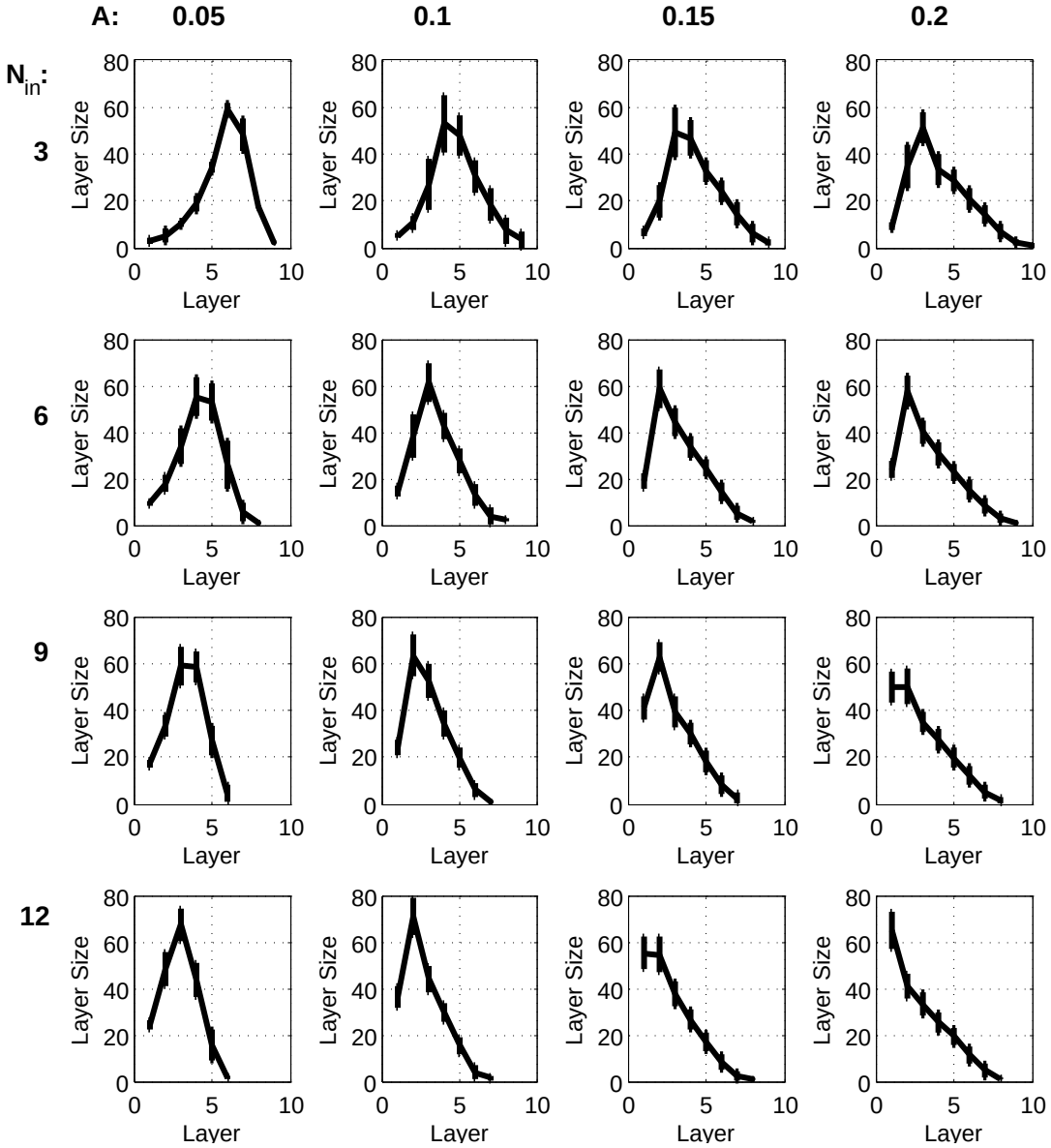


Figure 3.17: Effect of learning rate and input size. The length of the final chain and the width of the largest layer are dependent on the learning rate and input size. Error bars are standard deviation over 50 runs. Number of pool neurons N =200, learning rate A and number of input neurons N_{in} are given in figure labels. All other parameters are given in Table 3.1.

3.3.5 Learning Rate

The shape of the final network can also be modified by the leaning rate. Clearly, if each plasticity step is bigger, then fewer potentiation steps are needed to induce recruitment. Interestingly, with the parameters used for the simulations presented in Figure 3.17 this effect is not as straight forward as that seen for the increase in input size, seen above. Whilst the first layers follow an analogous pattern, with high learning rates giving larger first layers, this does not translate to shorter networks. In fact, amongst the higher leaning rates ($>$0.15) there is very little difference in network shape, particularly in the later layers.

The explanation for this counter intuitive effect comes from the interaction between the learning rate, A, the firing threshold θ and the layer sizes, L_N. When the learning rate is very high it requires only a few potentiation steps to recruit a neuron and when layer sizes are sufficiently large such that $L_{n-1} = \theta/A$, even a single potentiation step is sufficient to recruit a neuron. At this point the large layers dominate the recruitment and deplete the network pool before layer layers begin to form. Figure 3.17 illustrates the possible lengths and widths of networks given a range of learning rates and input sizes. A more thorough description of the interaction between input size, learning rate and network structure can be found in Chapter 5 where network development is described as a random walk.

3.3.6 Spontaneous Rate

In all the experiments previously presented, the spontaneous, λ_p, and input rate, λ_{in}, have been fixed at 0.1 Hz and 3 Hz respectively. However, if triphasic STDP is to be considered a candidate biological mechanism for synfire chain development, it must be robust to a wide range of parameters including the relative rates of the spontaneous and input firing.

Here we investigate this question by fixing λ_{in} at 3 Hz and running multiple experiments varying λ_p. The spontaneous rate was explored in the range 0.01 Hz to 100 Hz. There where ten runs for each value. As illustrated in Figure 3.18, increasing the spontaneous rate above the input rate has a detrimental effect on the chain development. With λ_p at 10 Hz, 90% of the simulations result in development of a chain structure. With λ_p at 100Hz, the interactions between the pool neurons dominate the plasticity events and no chains form. When λ_p is less than λ_{in} the network developed a chain structure every time. Thus, for successful development of synfire chains in this simple model system a more frequently firing input set is a strict requirement.

Also illustrated in Figure 3.18, decreasing the spontaneous rate increases the time it

takes to develop the chain. The explanation for this is simple: recruitment to the chain is the culmination of many potentiation steps, and each of those require a spike-pair. Reducing the spontaneous rate will reduce the rate of potentiation and therefore increase the time it takes for the network to develop.

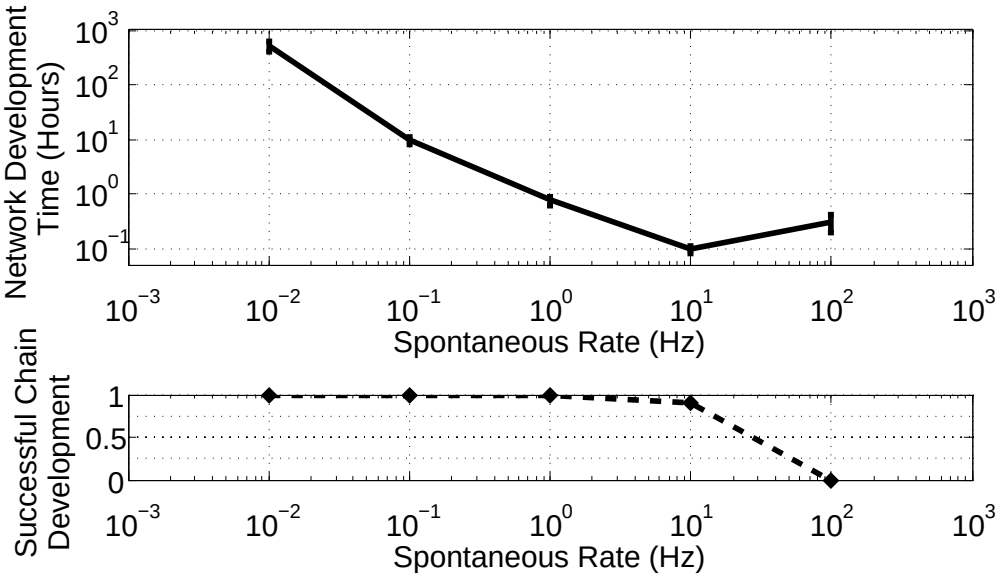


Figure 3.18: Effect of Spontaneous Rate. Simulations take longer when the spontaneous rate is low (top). The spontaneous rate must be below the input rate (3 Hz) to ensure chain development (bottom).

We saw above that altering the rate of recruitment can alter the resulting network shape. How then does the spontaneous rate, with its knock on effect on recruitment rate effect the final shape? Using only those simulations that resulted in chain growth, the average layer sizes for each layer are compared in Figure 3.19. The spontaneous rate appears to have only a small effect on the network shape, with lower rates shifting the balance of neurons downstream. There seems to be a countering effect once the spontaneous rate goes above the input rate. Overall, compared to other parameters, the effect is slight. This is because slowing the simulation through a reduction in spontaneous rate has an equal effect at each layer, so does not change the relative probability to recruit as the input size does.

3.3.7 Multiple Inputs

In the results described so far, there has been one distinct set of inputs projecting onto the remainder of the network. However, in the brain, there are often multiple inputs which compete with each other for control of target neurons. For example the projections from either eye segregate the target tissue into ocular dominance columns [88]. In songbirds

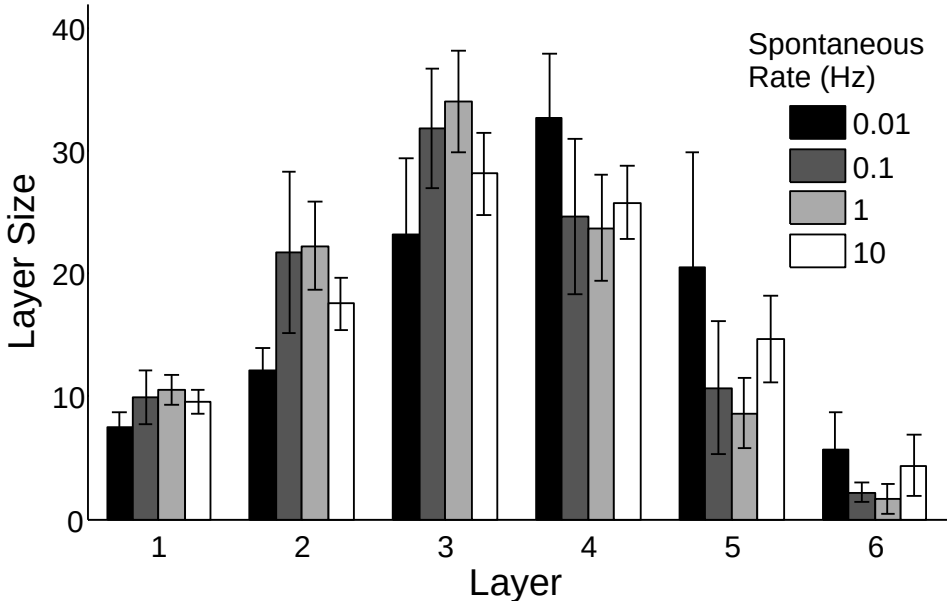


Figure 3.19: The effect of spontaneous rate on network shape. Error bars are standard deviation over 10 runs. All parameters are as in Table 3.1.

distinct synfire chains are thought to exist within HVC [98], each representing individual syllables of a song [55]. In the previous chapter it was found that multiple chains grown using classical STDP and pruning will merge together. However, two other authors have successfully embedded multiple chains within model networks [38, 96]. Both found that additions to classical STDP were necessary for multiple inputs. Fiete *et al* used heterosynaptic plasticity to ensure that chains did not cross over. Whilst Liu and Buonomano (2009) used presynaptic-dependent scaling. In this section we ask what happens if our system includes multiple inputs. Furthermore, how does the number of distinct inputs affect the final network configuration?

The addition of further input neuron groups was realised by segregating the input neurons into I distinct groups. The distinction between groups is one of firing time. Each input group consists of, N_{in}/I, neurons which fire synchronously at the rate λ_{in}/I Hz. Inputs are periodic with a rate of λ_{in} Hz, but, which input fires is randomly assigned. As a result, although the rates are constant the input firing is no longer strictly regular. The randomness is included to remove any bias that might occur from having the inputs occur in the same order each time. An example simulation with 3 inputs is presented in Figure 3.20.

Recruitment in the multiple input simulations progresses as before: synfire chains grow from the inputs. Now, however, recruitment can be to one of each of the input groups. These inputs now must compete for recruitment from the finite group of pool neurons. In every simulation that includes multiple inputs, multiple chains grow. How-

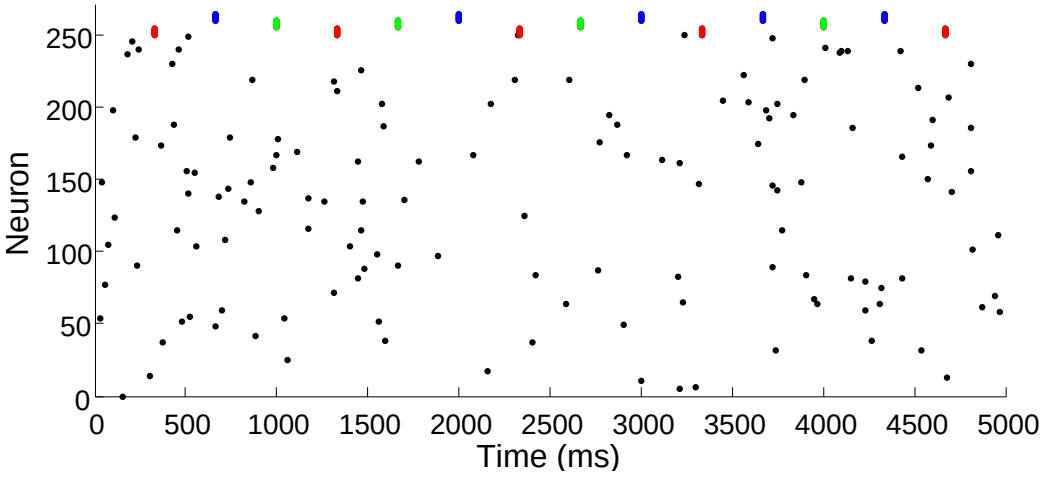


Figure 3.20: Network activity at the start of the simulation, 220 network neurons, 3 sets of 5 input neurons: input I=red, input II = green, input III = blue. All other parameters as in Table 3.1.

ever, the distribution of neurons between the chains is not equal.

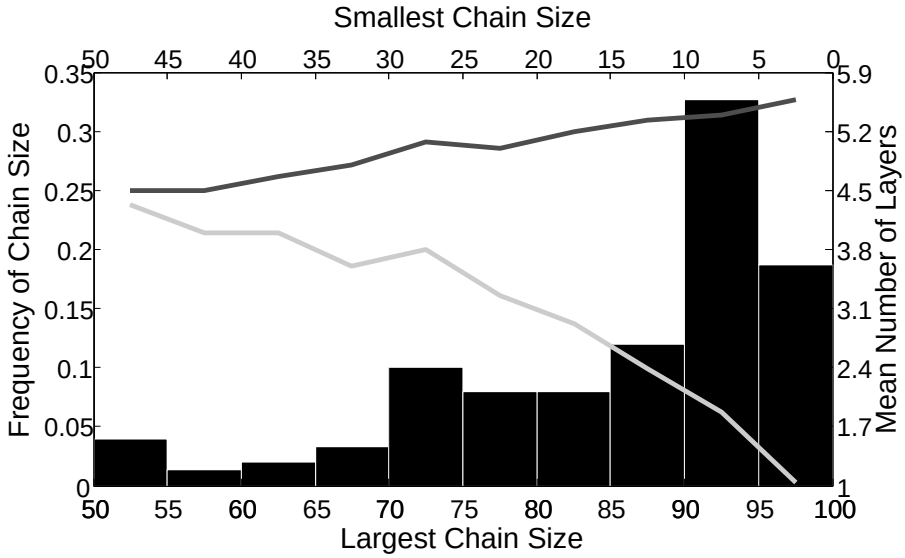


Figure 3.21: Histogram of Chain Size for Two Inputs. As the total number of neurons is constant (N = 100), each bar represents the frequency of both the smallest chain size (upper x axis) and largest chain size (lower x axis) from 150 runs. The difference in the lengths of the largest (dark grey) and shortest (light grey) chain is in keeping with the divergence in chain size.

With two inputs, it was usually the case that there emerged one dominate chain that grew larger and faster than the other. This phenomenon can be directly related to the effect layer size has on recruitment rates. Larger layers are able to recruit new neurons

faster than smaller ones. The initial symmetry breaking between the two chains occurs by chance, but once one of the chains becomes larger, this immediately leads to a positive feedback loop, which only intensifies as the chain sizes diverge. Figure 3.21 displays the distribution of chain sizes and lengths when the simulations contain two input groups. The most common result is one large (90-95 neurons from 100) chain and one (5-10 neurons) smaller one.

When the simulations are extended from one or two inputs to many, the dominant chain phenomenon persists. Figure 3.22 shows a typical development for a system with three inputs. Simulations with up to ten input groups are run and in each case there are usually as many chains as there are inputs. However, at least 60 % of the neurons belong to a single dominant chain (See Figure 3.24). In all cases the chains are independent of each other. There are no synaptic connections across chains and each neuron belongs to at most one chain.

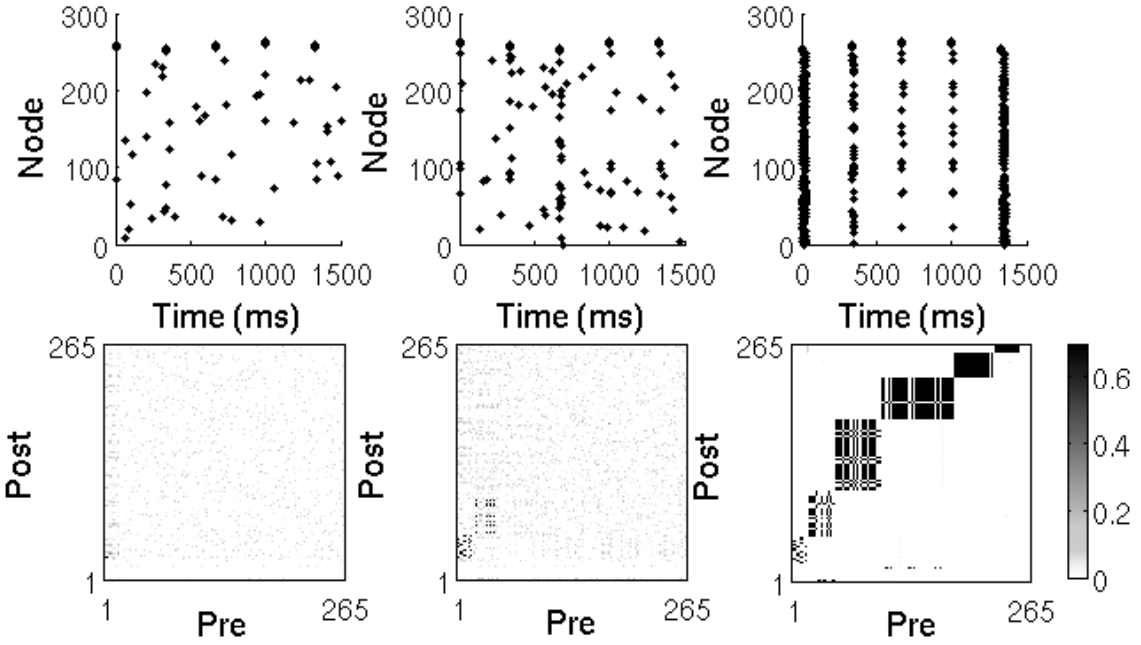


Figure 3.22: A three input simulation. $N_{\text{in}} = 5, I = 3, N = 250$. The first input group (neurons 250-254) is the first to recruit a node and continues to dominate the recruitment until finally recruiting the majority of the neurons.

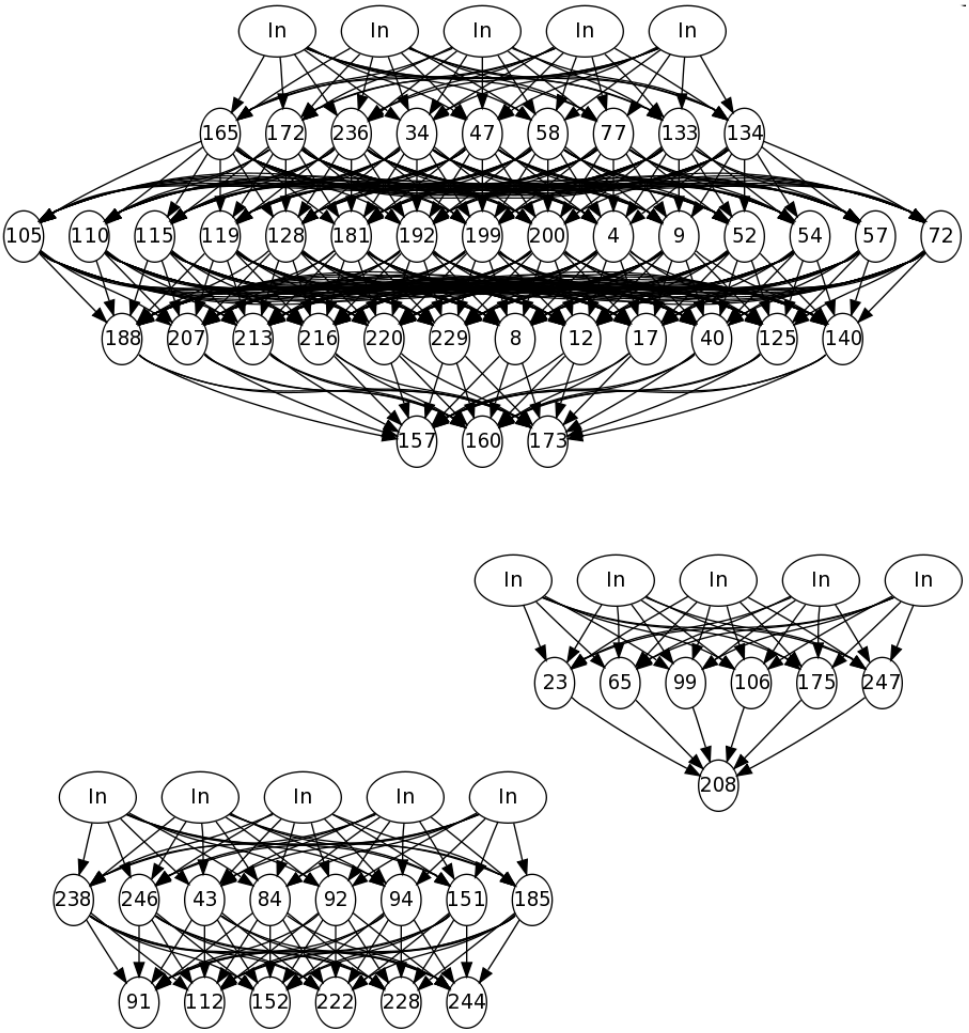


Figure 3.23: An example network of 250 neuron in which three independent chains grow from three inputs. Development is two thirds complete, same simulation as Figure 3.22

3.4 Conclusions

In this chapter, a novel approach to growing synfire chains in simulated networks was presented. In contrast to all previously published work, here synaptic plasticity was determined by a triphasic STDP rule applied alongside activity dependent excitability. The use of triphasic STDP removed the necessity to impose topological constraints on the developing neural circuit. In previous work, strong topological constraints have dominated development, producing strictly regular chains with predetermined layer widths [38, 77]. In contrast, with triphasic STDP, the final structure is determined through a competitive

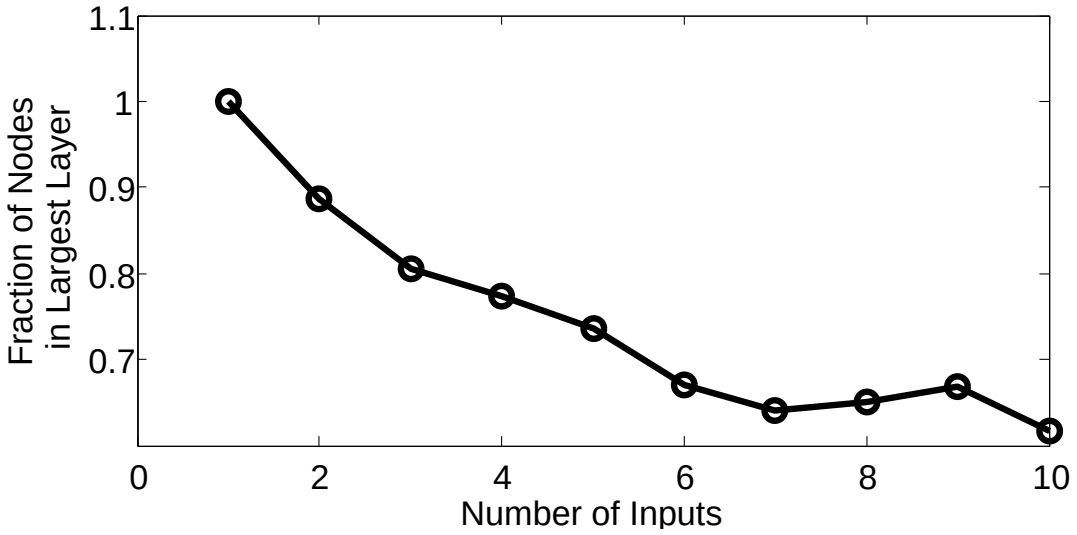


Figure 3.24: Fraction of neurons that are recruited to the largest chain. Even with ten competing inputs, more than 60% of nodes are recruited to one dominant chain. Mean over 10 runs.

recruitment process guided by the network's initial conditions and the STDP function.

For successful chain development, the system must meet four criteria. Firstly, the STDP rule must induce potentiation for short positive spike-time-differences and induce depression for long positive and negative spike-time-differences. The crossing point from potentiation to depression should occur after one transmission delay, but before two. Secondly, spontaneous activity must occur within the developing network and desist once neurons are recruited to the chain. Thirdly, synapses which receive uncorrelated (defined here as large Δt) spikes, should be slowly depressed. Finally, for robust development, the spontaneous activity must have a rate lower than that of the input group.

In this investigation the parameter, α, of the STDP function was set to 4 ms, thus any spike-time difference greater than 8 ms will result in depression. This agrees with the experimental observation in ref. [32], but is less than the observed duration in refs. [116] and [167], where depression was reported at spike-time differences greater than 15 ms and 25 ms respectively. This model depends on a strict relationship between the time scales of the triphasic STDP function and the transmission delay. The duration of transmission is not known for the above experimental preparations. Experimental support for this model would include, not just further evidence of triphasic STDP, but also evidence that the time scales of the triphasic STDP function and the time scales of transmission in the given

preparation were aligned. Furthermore, the preparation should be from an area thought to contain synfire chains, HVC of song birds, for example [98].

The goal of the modelling work in the present chapter was the development of feed-forward structures, specifically, synfire chains. Any recurrence in the structure was critically damaging for the model and lead to collapse of the chain. The brain is not an entirely feed-forward structure; highly recurrent structures are thought to be important in working memory [17] and feed-forward structures like synfire chains, may be embedded within recurrent systems for realisation of more complex behaviours [55]. This simple model of chain development would need to be augmented to allow such desirable recurrent connections whilst removing others. Such supplementary work may include a range of neural types and corresponding plasticity rules.

In addition to triphasic STDP, this model relies on the suppression of spontaneous activity through activity dependent excitability. This assumption is experimentally motivated [109, 150, 154, 165]; nonetheless, neural systems are famously noisy and if triphasic STDP is employed in biological systems, it would need to be robust to random spiking events. In this model, when layers become large, a single spontaneous spike can lead to the formation of new connections. This is unrealistic and a result of the very simple model presented here. It may be the case that more realistic networks, which do not fire in perfect synchrony, which have sparse connectivity and much slower learning are more robust to isolated spiking events.

In conclusion, triphasic STDP was shown to grow synfire chains in a network of binary neurons. Using this simple neural model it was possible to investigate the workings of this plasticity rule unobstructed by the highly complex behaviour observed in networks constructed from more realistic neural models. However, for the triphasic STDP hypothesis to be considered strong, it should also be demonstrated in a more realistic model and this is the subject of the next chapter.

Chapter 4

Growing Synfire Chains without Topological Constraints: A Leaky Integrate and Fire Model

---

The results in Chapter 3 show that triphasic STDP can be used to grow synfire chains in a network of binary neurons without additional topological constraints. Nonetheless, this plasticity rule has not been tested with more realistic neural models. It was shown that classical STDP can grow synfire chains when topological constrains are added regardless of the neural model. However, it may be possible that the simplicity of the binary model masked a conflict that might exist between triphasic STDP and more realistic models. For example, in conductance based models, non-instantaneous synaptic inputs, coupled with the slow leak currents enable summation of asynchronous inputs, this cannot occur in a binary model. To validate the results obtained using the binary model and therefore strengthen the hypothesis that triphasic STDP could be employed for synfire development, here development through triphasic STDP is implemented in a more biologically realistic neural model. Firstly, the leaky integrate and fire (LIF) model is introduced. Next, the similarities and differences between this and the binary model are discussed. Finally, those parameters specific to this model are explored.

4.1 The Model

This network development model uses leaky integrate and fire (LIF) neurons, which unlike binary neurons are able to aggregate inputs over time giving more realistic neural behaviour. The neural simulation software NEST [101] (version 2.0-0-rc2) was used to undertake this modelling work. The neural model was chosen from those distributed with NEST ("iaf_cond_exp"). In addition, a bespoke synaptic model was written ("tri_synapse") which extended the simulator, this can be found along with the simulation scripts on the accompanying CD.

4.1.1 The Neural Model

As in previous models, the network consists of N identical network neurons and N_{in} input neurons. Each input neuron has excitatory efferent synapses onto every network neuron, but receives no afferent synapses from the network. The network neurons are completely recurrently connected with excitatory synapses excluding self connections. The change in membrane potential, V_j, of neuron j is given by

$$C\frac{dV_j}{dt} = -g_L(V_j - E_L) - g_{\text{syn}}(V_j - E_{\text{ex}}) \tag{4.1}$$

where C is the membrane capacitance, E_L and E_{ex} are the reversal potentials of the leak and excitatory synaptic currents respectively, g_L is a constant leak conductance, and g_{syn} is an exponential synaptic conductance given by,

$$\frac{dg_{\text{syn}}}{dt} = -\frac{g_{\text{syn}}}{\tau_{\text{syn}}} + \sum_i W_{ij}s_j(t-d) \tag{4.2}$$

where τ_{syn} is the synaptic time constant; W_{ij} is the weight of the synapse and $s_j(t)$ is 1 if neuron j fired at time t and 0 otherwise; d is the delay between neurons.

If the membrane potential V_j reaches the firing threshold, T, the neuron is said to have spiked and V_j is reset to the reset potential, V_{reset}. The neuron then enters an absolute refractory period of t_{ref}, where V_j is fixed at V_{reset}.

Table 4.1: Model Parameters

	Simulation Parameters
N	100
N_{in}	5
d	5.0 ms
λ_p	0.1 Hz
λ_{in}	3 Hz
W_{max}	20.25 nS
A	9.5
τ_{syn}	0.2 ms
V_{th}	-50 mV
t_{ref}	20 ms
E_L	-85 mV
V_{reset}	-80 mV
V_m	-80 mV
E_{ex}	0 mV
g_L	1.125 nS
C_m	22.5 pF

4.1.2 Synaptic Plasticity

The weights of intra-network excitatory synapses, W_{ij}, are modified according to the triphasic STDP rule. The change in synaptic weight, ΔW, is given by

$$\Delta W_{ij} = A\left[1 - \frac{(\Delta t_{ij} - \alpha)^2}{\alpha^2}\right] e^{\frac{-|\Delta t_{ij} - \alpha|}{\alpha}}, \tag{4.3}$$

where Δt_{ij} is the time difference between spikes in neuron i and neuron j ($\Delta t_{ij} = t_j - t_i$); α is a scaling parameter which determines the width of the potentiation window (the function crosses from potentiation to depression at $\Delta t_{ij} = 2\alpha$) and A is the learning rate. The synaptic weight is bounded by 0 and g_{max}. The rule is additively applied between nearest neighbour spikes only. All simulation parameters are as is in Table 4.1 unless otherwise stated.

4.1.2.1 Spontaneous Activity

As in previous simulations, network neurons fire stochastically. Here this is induced through a single Poisson input to each neuron with rate λ_p which has fixed suprathreshold excitatory connections such that a single spike causes a further spike in the receiving neuron. This implementation, in contrast to the collection of balanced inputs (Chapter 2), does not result in a fluctuating membrane potential. When network neurons are recruited

(defined as firing within 1 Hz of the input rate) λ_p is reduced to zero. I.e.,

$$\lambda_p(\lambda_j) = \begin{cases} \lambda_p & \lambda_j < \lambda_{\text{in}} - 1Hz \\ 0 & \text{otherwise} \end{cases} \tag{4.4}$$

where λ_j is the firing rate of neuron j. Input neurons do not receive background input but fire periodically at rate λ_{in}.

4.2 Results

Just as in the network of binary neurons, triphasic STDP generates synfire chains in a network of LIF neurons. Figures 4.1 and 4.2 show an example of the final network structure following development. In this network of 100 LIF neurons five layers develop, the third layer being the largest. Again, as in the binary model, neurons are continually added to all layers during development, with the growth rate dependent on the previous layer size and the number of available neurons (Figure 4.3). This development trajectory is very different to that seen with classical STDP and topological constraints (Chapter 2). There layers form one by one, with each layer forming relatively fast compared to the time taken for the entire chain to grow.

4.2.1 Comparison to Binary Neurons

4.2.1.1 Refractory Period

In Chapter 2 a network of LIF neurons was described. There, synfire chains grew through a combination classical STDP and topological constraints. In that model, it was necessary to set the refractory period, t_{ref}, of the neurons to at least three times the delay, d. When $t_{\text{ref}} \leq 3d$ the structure developed loops. With an insufficient refractory period, backwards connections were maintained in the network. The resulting activity pattern was self sustaining and repeating with period equal to t_{ref}. Conversely, in Chapter 3 a network of binary neurons was described. There, as here, synfire chains grew through triphasic STDP plus activity dependent excitability. In the binary model it was sufficient to set $t_{\text{ref}} > d$. Here triphasic STDP is used to grow synfire chains in a network of LIF neurons. Therefore, it is not clear what duration the refractory period must have.

To address this question, simulations were run with refractory periods in the range 3 ms to 24 ms. The results of the simulations are presented in Figure 2.9. In the newest implementation, the refectory period must be greater than d (set at 5 ms). The previous

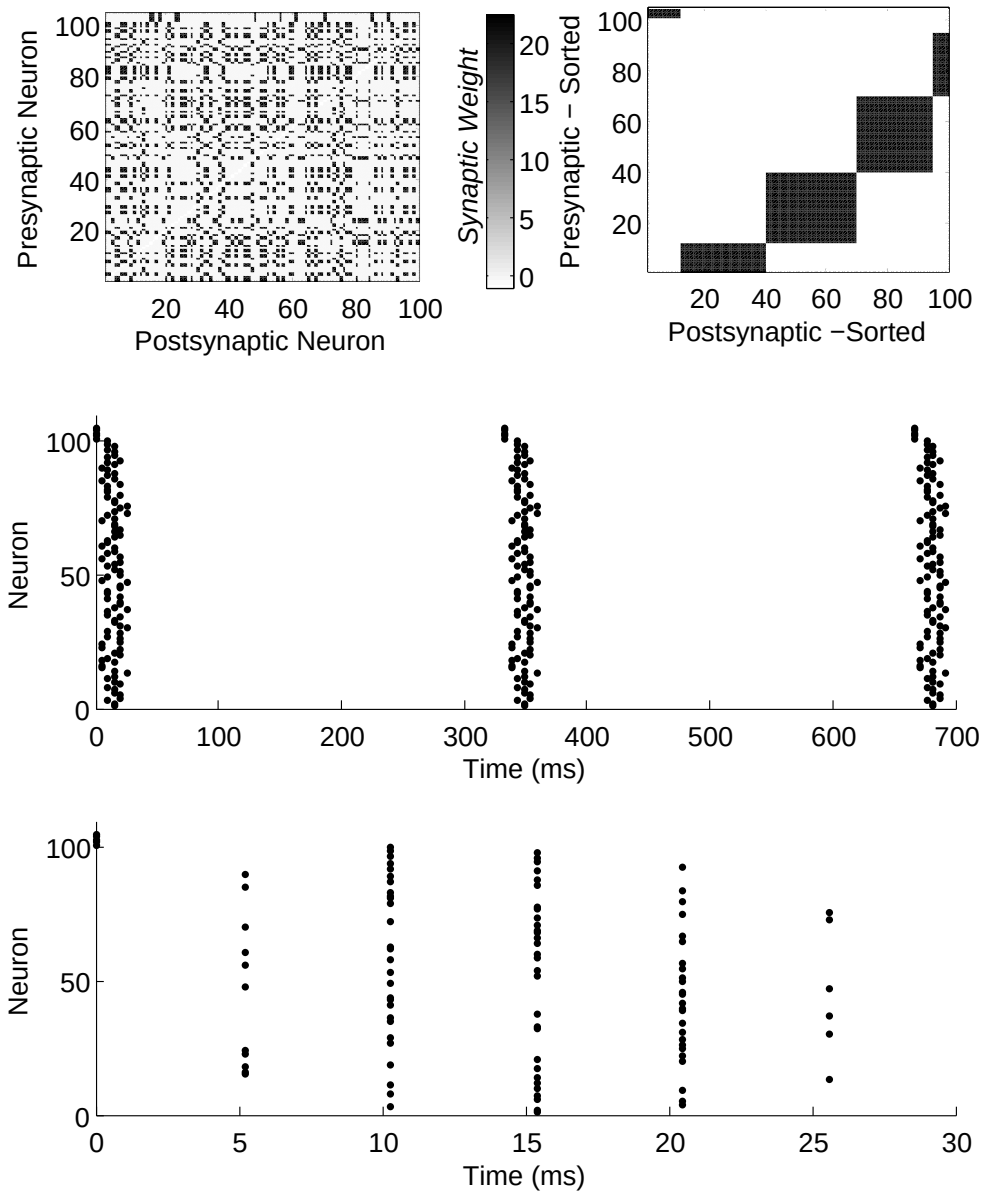


Figure 4.1: The final structure of the network is displayed, with the neuron IDs sorted by spike time difference from the input (top right), and as originally assigned (top left). The raster plot (middle) clearly shows a repeating structure. The raster plot (bottom) illustrates the layer structure, with spikes occurring in groups exactly 5 ms (d) apart.

restriction of three times d is not required. The ability to shorten the refractory period is a major benefit of development with triphasic STDP (Although at 6 ms, this may still be considered unrealistic). A requirement for parameters outside of the physiological range can shed doubt on the validity of any computational model and is a major drawback in the classical STDP plus topological constraints model (Chapter 2 and Ref. [77]).

Five simulations were run with t_{ref} set at three milliseconds. Of these, four failed to

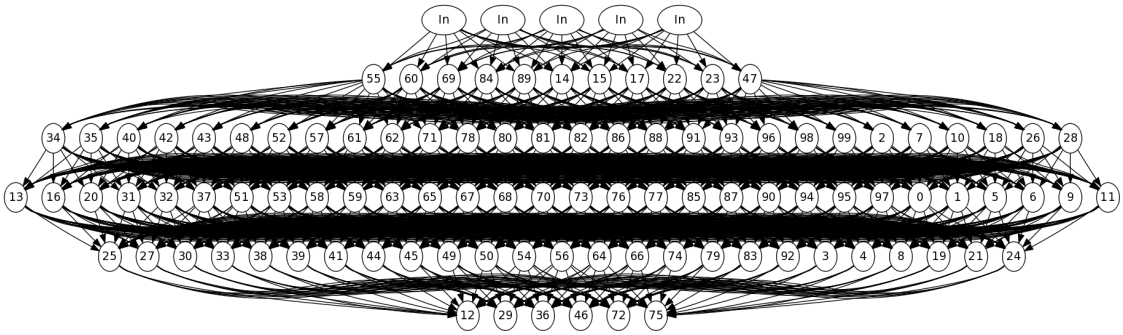


Figure 4.2: The final chain structure. As in Chapter 3 The network is entirely feed forward.

develop chains. Instead, a repeating pattern can be seen in the network, with all neurons firing at saturation rates (Figure 2.9). The fact that failure was not universal suggests that it is the result of a stochastic effect. The only random element of these simulations is the timing of spontaneous activity, thus a misplaced spike must have lead to the erroneous network development. That is exactly the case; a spontaneous spike in a neuron not yet mature but with connections onto the now well developed chain caused backward connections to develop from the last to the penultimate layer. As there are already many nodes within the layer, these backward connections accumulate instantly. Because the neurons have already exited their brief refractory period, the layer fires again. This quickly established loop is self-reinforcing and leads the collapse of the entire chain.

Of the simulations where t_{ref} is greater than d there is little (Figure 2.9) difference between them in either the width or length of the network or the development duration. However, those simulations in with t_{ref} at 24 ms did take, on average, a longer time to recruit the first neuron. A long refractory period will reduce the firing rate of the network neurons as a greater fraction of the incoming events will not result in spikes. This lower spontaneous rate, reduces the probability per time step of a recruitment, thus, it will increase the average time until the first recruitment.

These experiments demonstrate that unlike synfire growth with classic STDP (chapter 2 and Ref. [77]), stable chains can be grown through triphasic STDP and activity-dependent excitability with an LIF neural model and shorter refractory periods. Next, the learning rate element of the STDP rule is addressed.

4.2.1.2 Learning Rate

The learning rate, A, determines the magnitude of plasticity steps in the simulation. Naively, one may wish to set the learning rate (i.e., maximum STDP step) as a fixed

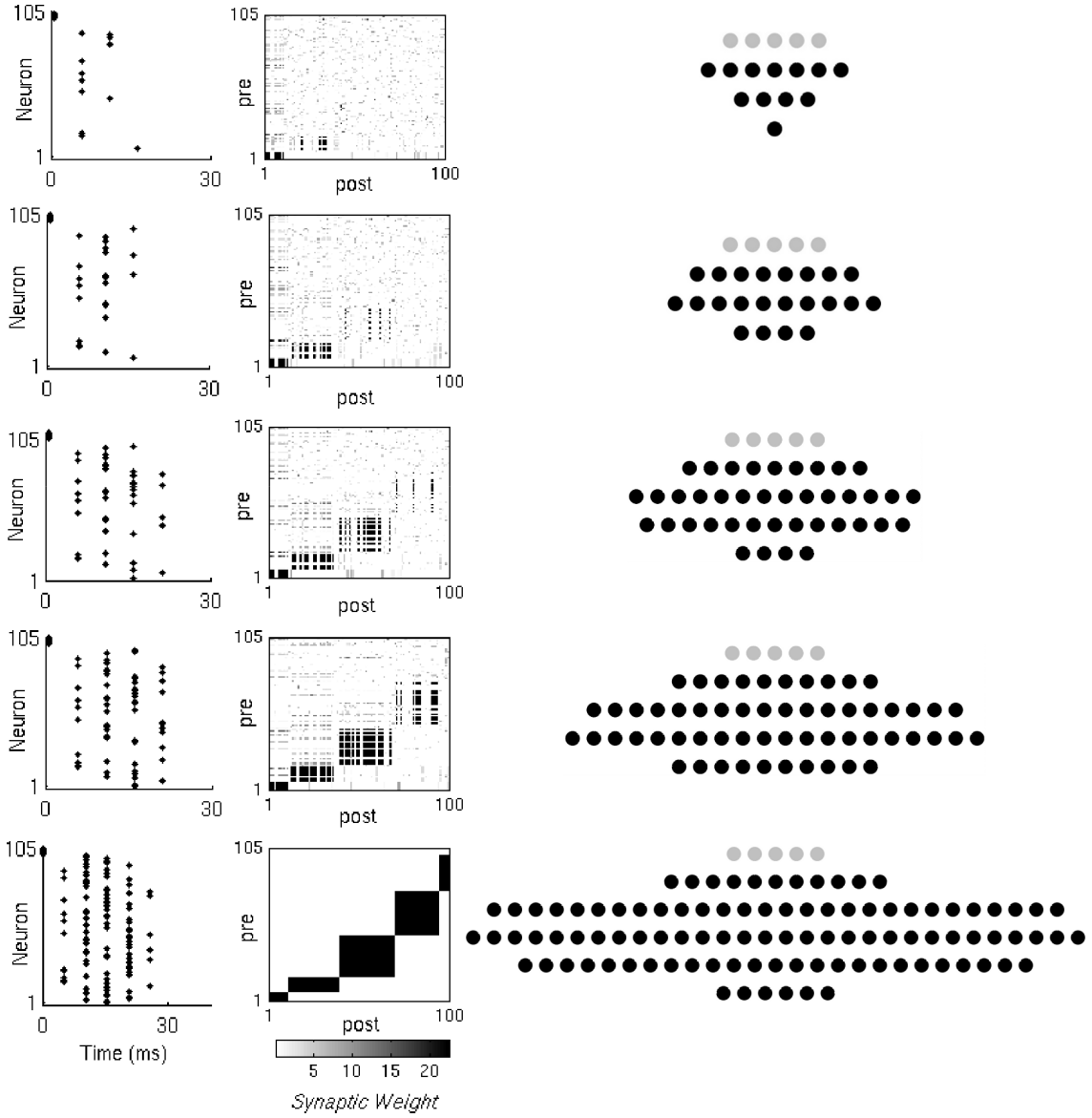


Figure 4.3: Network Activity and Synaptic Weights. Network activity (left), weight matrix evolution (centre) and network structure (right) during synfire chain development (from top to bottom). Neuron indices in weight matrices have been sorted by spike time order (in response to the last input in the simulation), and thus represent the neuron's position in the developed chain.

ratio of the maximum weight (or synaptic conductance $W_{\max}$). For example, in the binary model $\frac{A}{W_{\max}} = \frac{0.1}{0.66} \approx 0.14$. Here, the maximum synaptic conductance is set at 20.25, 0.14 of which is just less than 3. However, synaptic weight has a slightly different meaning in the two different neural models. In the binary model, a synapse with weight W increases the postsynaptic neuron's membrane potential by W. In the LIF model W is the peak conductance of the synapse. The effect on the membrane potential of the receiving neuron is

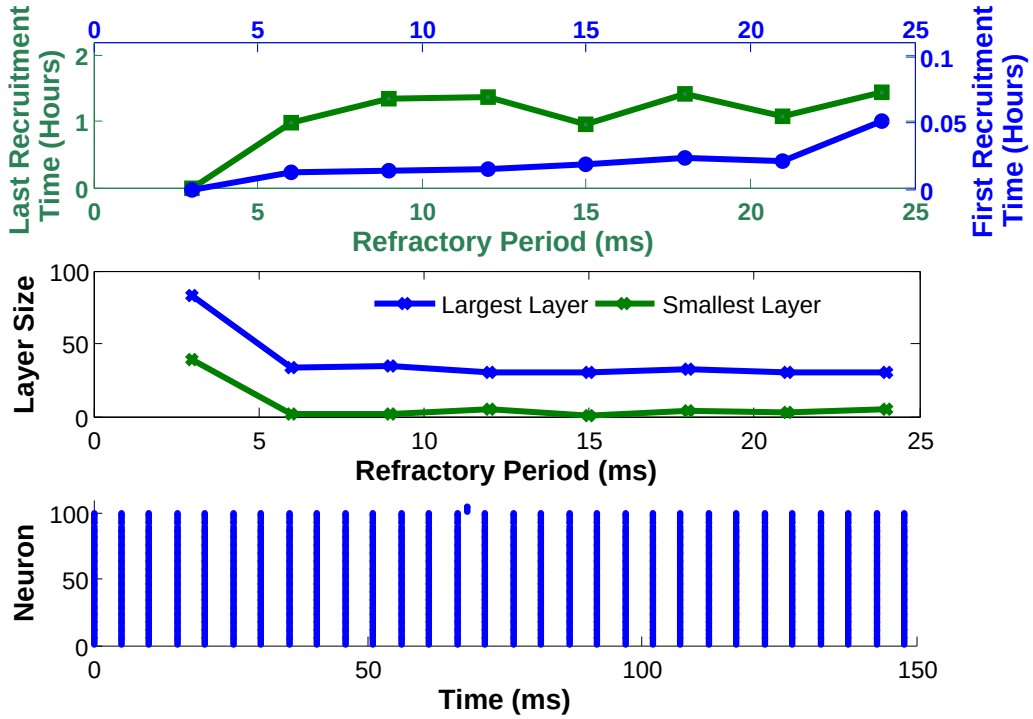


Figure 4.4: The effect of refractory period on mean recruitment times (top) and mean layer size (middle). Raster plot of collapsed network activity when refractory period set to 3ms. Data from single runs at each t_{ref}.

additionally determined by the time constant of the synapse (τ_{syn}), the reversal potential of the synapse (E_{ex}) and the current membrane potential (V). τ_{syn} and E_{ex} are static and defined at the start of the simulation whereas the membrane potential V is dynamic and changes throughout the simulation. Due to this complication, it is not possible to map the learning rate exactly from the binary model to the LIF model.

In the absence of a direct translation between the two models, simulations with a range of learning rates values were conducted. The resultant network structures are presented in Figure 4.5. As in the binary model, the learning rate modifies the network structure. Small values for A give smaller first layers and a later peak; the largest layer being layer four. Larger values of A give longer chains (ten layers instead of nine), larger early layers and an earlier peak; the largest layer being layer two. The value of learning rate which most resembles structure produced by the binary model is $A = 9.5$. This value was then used to compare the two simulation types.

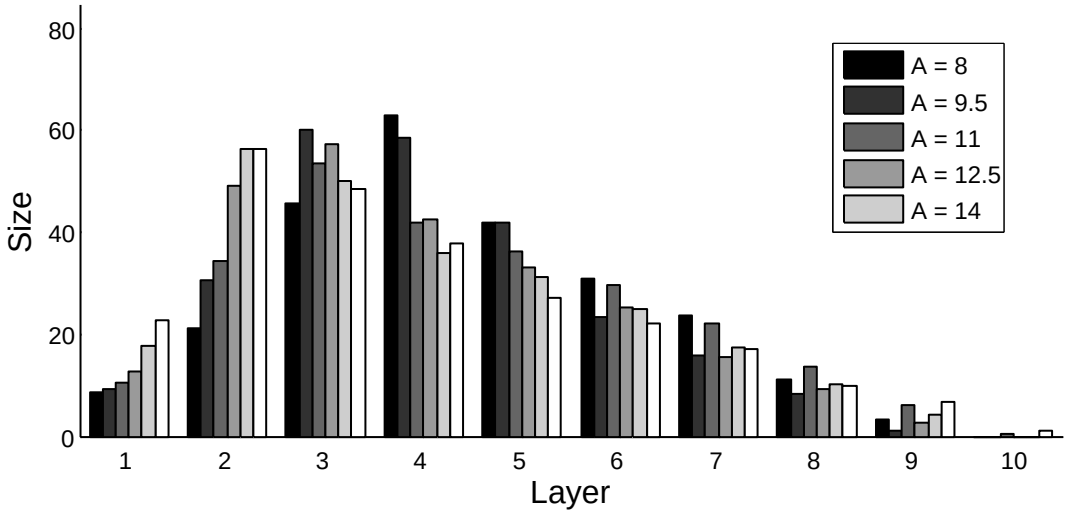


Figure 4.5: Mean layer sizes with various learning rates. Altering the learning rate effects the final layer distribution. Networks had 250 network neurons and 5 input neurons, mean is of 5 runs.

4.2.1.3 Qualitative Comparison of Binary and Integrate and Fire Models

The stochastic nature of these simulations results in differences between individual runs even when all parameters are the same. Hence, to better understand the difference between the two neural types (binary and LIF), it is necessary to compare the statistics from many runs of each type. Here, LIF simulations with A set at 9.5 were compared to simulations from the binary model. Each network had 250 network neurons and 5 input neurons, all other parameters were as Table 4.1. All comparisons in this section are from 100 simulation runs.

Figure 4.6 shows the final network structure from each of the neural types. The LIF model produces longer chains with the largest layer being slightly smaller. This difference in network structure is attributed to the dynamic learning rate described above. Whilst plasticity steps are uniform for a given simulation in the binary model, they are dependent on the current state of the neuron in the LIF model and effectively reduced towards the end of the simulation. As a result, the final network structure is modified towards what one would expect for slightly lower learning rate (smaller first layer and longer chain). Note however that simply reducing the learning rate in the LIF simulation would not bring these models together. As stated above, it is not possible to exactly match the learning rates in the two models. Allowing for the slight difference in learning rate, the network structures produced by each model are in fact very similar, both displaying the characteristic diamond shape with the peak at layer three.

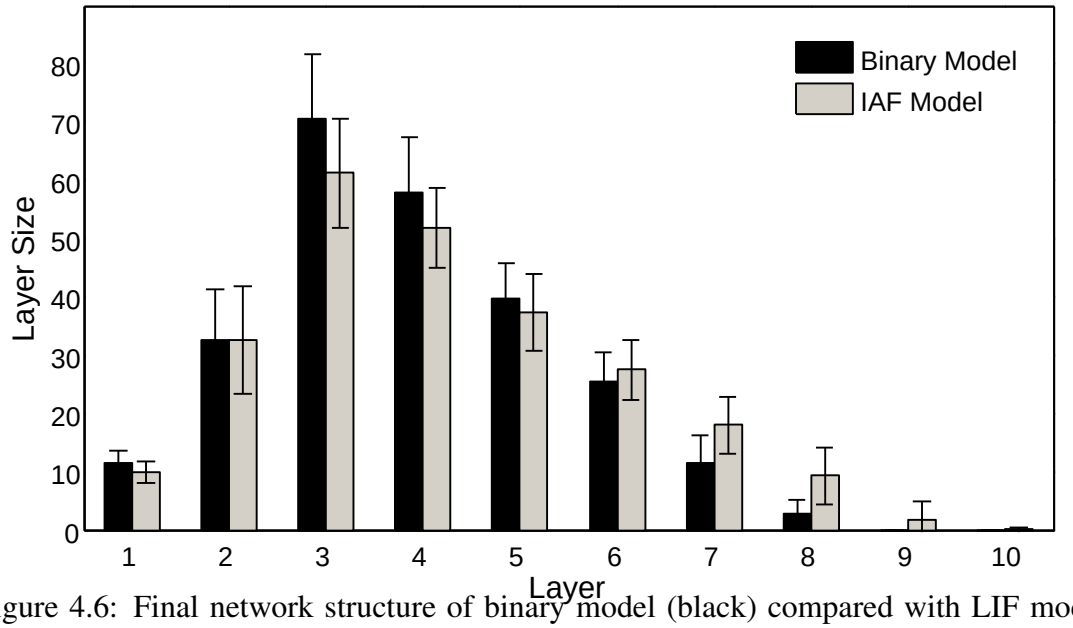


Figure 4.6: Final network structure of binary model (black) compared with LIF model (grey). Error bars show standard deviation.

Recall that how the STDP rule is applied (additively or multiplicatively) effects the final distribution of weights (Section 2.1.3.5). Here we examine whether the dynamic learning rate described above has an effect on the distribution of weights in these simulations. Figure 4.7 shows that whilst the majority of weights in both models are at either extreme, in the mid-range the LIF model displays a tendency towards lower weights. Conversely, the binary model is almost flat, with a slight tendency towards higher weights. The result is almost negligible, but is another illustration of the reduction in learning rate for higher weighted synapses. This tendency towards lower weights when using conductance based synapses may have been one of the reasons multiplicative STDP was required in the conductance based model of Ref. [38].

Figure 4.8 displays the final network activity of the two model types. Note how both display eight layers of synchronous activity following the input, with the largest layers in the middle of the activity pattern. However, the model types differ in the transmission time between the layers. The binary model has exactly 5 ms (the delay, d) between each layer. Whilst the LIF model has 5.1 ms (the delay plus 0.1 ms). The explanation for this rests with the neural dynamics of each model. The time step in the LIF model is 0.1 ms and spikes that are received at time t may cause a spike to be sent at some time later. The time, which depends on the synaptic dynamics (W, E_{ex} and τ_{syn}) will always be t plus a multiple of the time step. Thus when synaptic weights are strong, the total time between layers will be $d + 0.1$ ms. The binary model is event driven, hence has no time step at all. This small difference in transmission time, will slightly affect the size of the plasticity step and may have contributed to the slight difference in network structure observed (Figure

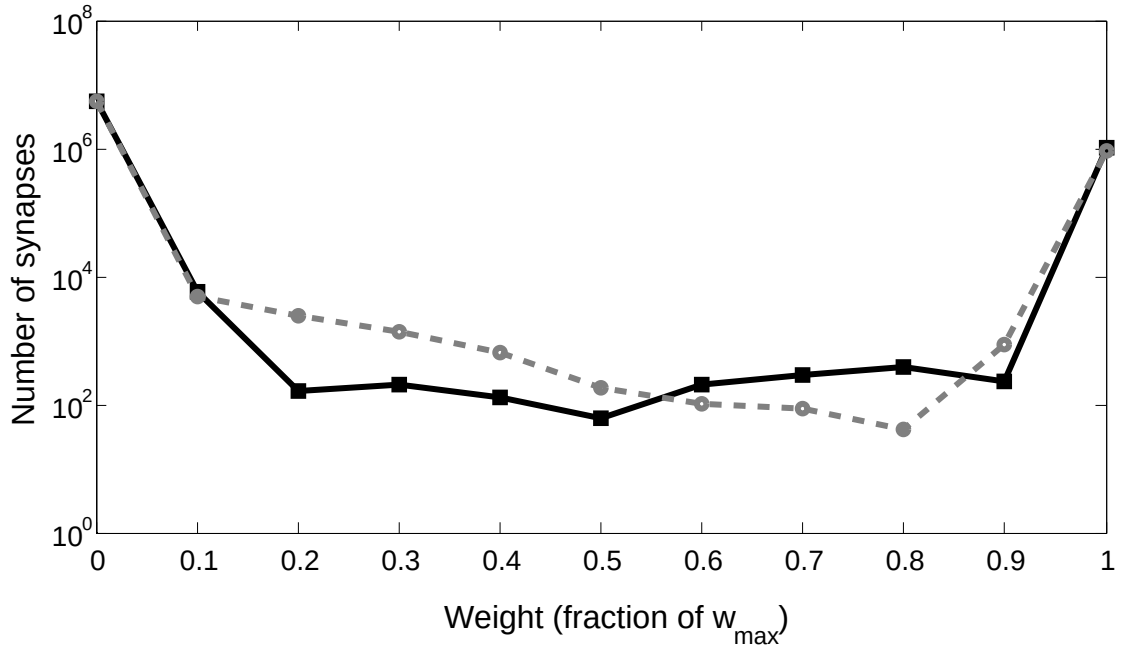


Figure 4.7: Final weight distribution of binary model (black) compared with LIF model (grey).

4.6).

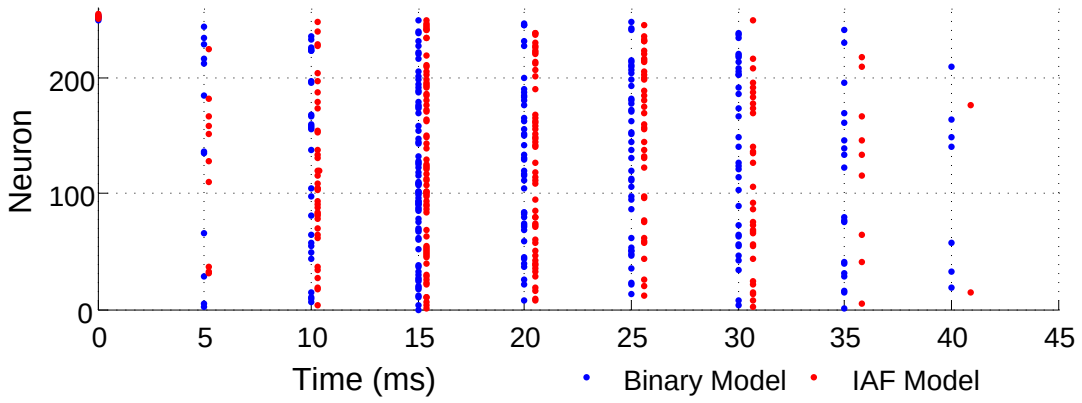


Figure 4.8: An example of final network activation of binary model (blue) compared with LIF model (red).

The final comparison between the binary and LIF model is the time course of recruitments. Figure 4.9 shows the mean and standard deviation for each recruitment event calculated from one hundred runs of each neural type. (I.e., the n^{th} recruitment time from all simulations is averaged and plotted for $n = \{1 \ldots 250\}$. The figure illustrates that the LIF model takes longer to recruit the entire network. Also, perhaps more interestingly, it shows that the recruitment trajectory is altered. In the LIF model, there appears to be a

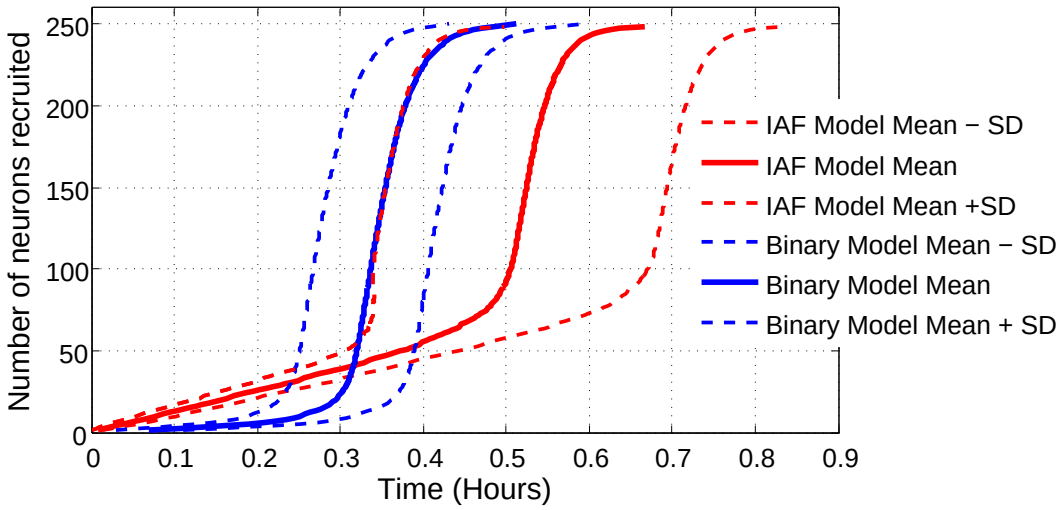


Figure 4.9: The mean recruitment times over 100 runs for binary model (blue) and LIF model (red). The dashed lines are plus and minus standard deviation

period of linear growth that begins earlier, is faster and lasts longer than the much shallower and shorter linear growth period in the binary model. The binary model moves from a period of relatively slow growth to a period of much faster growth at around 0.3 hours (approximately 20 neurons recruited). The LIF model reaches this inflection point much later, 0.5 hours (approximately 90 neurons recruited). After the respective inflection points, the recruitment rates are equal at 50 neurons per minute (Figure 4.10). Towards the end of the simulation, the rates drop off due to finite size effects (the finite pool of neurons N is depleted). For analysis into recruitment rates and the resultant network structures see Chapter 5.

This concludes the comparison of binary and LIF models. The above analysis has demonstrated that the move to a conductance based LIF model does result in interesting changes to the experimental results. These changes for the main part are due to the variable effect of synaptic weight and consequently plasticity step size in this model. The changes in results however are quantitative not qualitative; networks of LIF neurons grow diamond shaped synfire chains through triphasic STDP and activity-dependent excitability. As a result, it can be concluded that synfire chain development through triphasic STDP does not rest on the particular choice of neural model. This, in turn, strengthens the case that this is a possible mechanism by which synfire chains could develop in the brain.

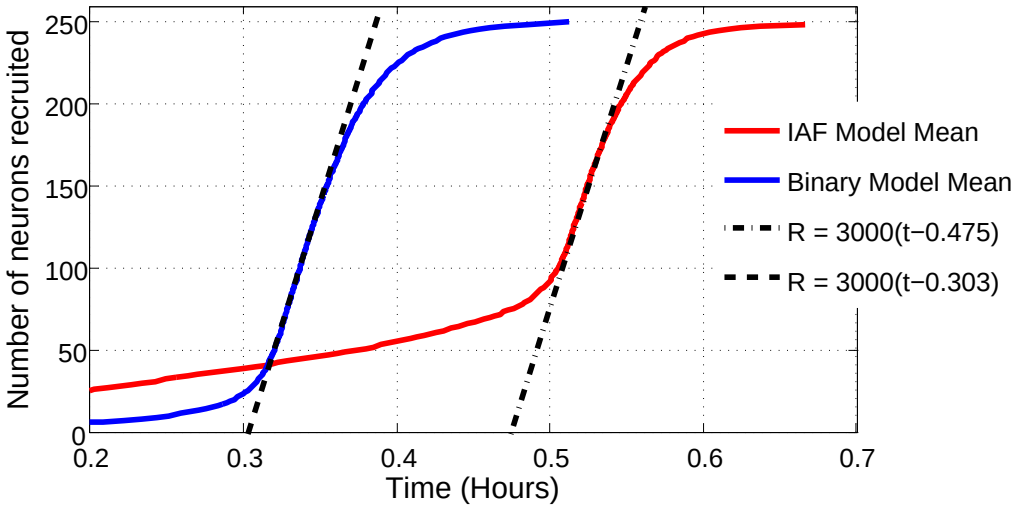


Figure 4.10: The peak recruitment rate is approximately 3000 neurons per hour (50 per minute) for both models

4.2.2 Parameters Specific to LIF Neurons

Having compared the two neural models with equivalent parameters, it is now appropriate to investigate the effect of varying those parameters that are specific to the LIF model. In addition, the flexibility of the NEST simulator allows networks to be constructed with a range of delays. Here the importance of the synaptic time constant and the distribution of transmission delays are investigated.

4.2.2.1 Synaptic Time Constant

Here synaptic inputs are modelled as jumps in conductance (of magnitude W_{ij}) which decay with time constant τ_{syn} (Equation (4.2)). This spreading of the synaptic input, coupled with the slow leak currents, enable summation of asynchronous inputs. This is the main advantage of the more realistic LIF model over the simpler binary model used previously. In the previous section, the parameters were based on those in Chapter 2 (τ_{syn} = 0.2 ms). However, depolarising excitatory postsynaptic potentials in HVC of zebra finch (a model system for synfire chain research) have been reported to be much longer (9ms to peak conductance) [112]. To understand how synaptic inputs of long duration affect the development of synfire chains, simulations were run with τ_{syn} between 0.2 and 12.8 ms. Simply increasing τ_{syn} will increase the total current received from a synapse with the same weight (Figure 4.11). Hence we modified the maximum weight W_{max} and the learning rate A (to maintain the ratio $W_{\text{max}} : A$) so that a network neuron fires a single spike upon receiving three incoming spikes at W_{max} (Figure 4.11 and Table 4.2). Without

the modifications to synaptic weight, long time constants result in multiple spikes, t_{ref} (20 ms) apart.

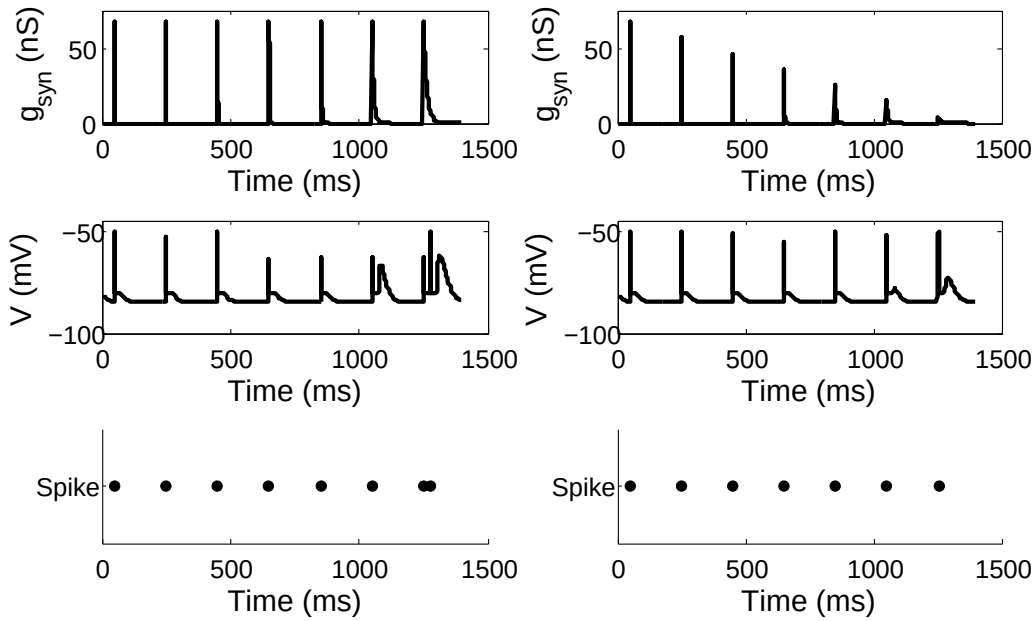


Figure 4.11: Synaptic time constant doubling with each spike from 0.2 ms. Left: with other parameters held constant, note two spikes t_{ref} (20 ms) apart when τ_{syn} = 12.8 ms. Right: with maximum weight modified as in Table 4.2.

Table 4.2: Parameter Adjustments with Synaptic Time Constant

τ_{syn} (ms)	W_{max}	A
0.2	22.5	9.0
0.4	19.0	7.6
0.8	15.5	6.2
1.6	12.0	4.8
3.2	8.5	3.4
6.4	5.0	2.0
12.8	1.5	0.6

In the simulations with τ_{syn} set to 6.4 ms or 12.8 ms chains grow but are not stable. Part way through development, loops are generated and the networks collapse to the now familiar repeating pattern with neurons firing at saturation levels. When the synaptic time constant is very long the neuron is still receiving synaptic input when it is released from its refractory period, the result being two spikes in quick succession. The layer now spikes both in its defined position and further down the chain. This results in the formation of erroneous connections, which result in loops and network collapse. When τ_{syn} is set to 3.2 ms or less, chains develop normally.

Figure 4.12 shows the relative layer sizes in the final chain structure for the range of τ_{syn} values with which chain development was successful. Larger values of τ_{syn} typically lead to longer chains in which the largest layer is relatively early. When τ_{syn} equals 1.6 ms, the first layer is the largest. Smaller values of τ_{syn} lead to shorter chains with smaller earlier layers.

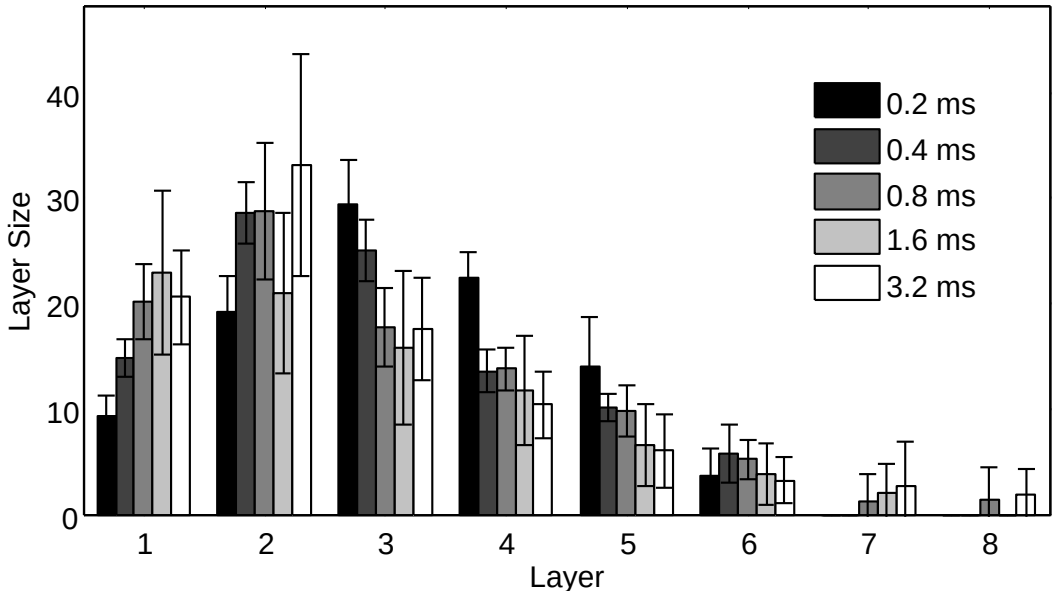


Figure 4.12: The effect of increasing the synaptic time constant on network shape. Mean and standard deviation of 5 runs plotted.

Figure 4.13 shows the network activity rate for the range of τ_{syn} values. The total simulation time is much longer when synapses are active for a short duration. The majority of this time however is spent recruiting the first few neurons. All simulations appear to follow similar trajectories once the rate reaches approximately 70 Hz. Once this rate is reached, there are at least 23 neurons in the network, the largest layer will contain around 10 neurons (see Figure 4.3). Recall that layers fire in synchrony, which results in identical STDP between any unrecruited node and all the members of a layer. If a layer contains 10 neurons then, following plasticity, the increase in stimulation for a downstream neuron is ten times the learning rate, A. In this situation, one potentiation step will be sufficient to recruit a neuron. Hence the recruitment rate is driven by the spontaneous activity and number of remaining pool neurons. In this regime, altering the synaptic time constant below 3.2 ms, has no effect. In the early stages of the recruitment process, it takes more steps to induce recruitment. In this regime, the changes to τ_{syn}, W_{max} and A are able to slow down the recruitment process.

Development of synfire chains through triphasic STDP is robust to a range of synaptic

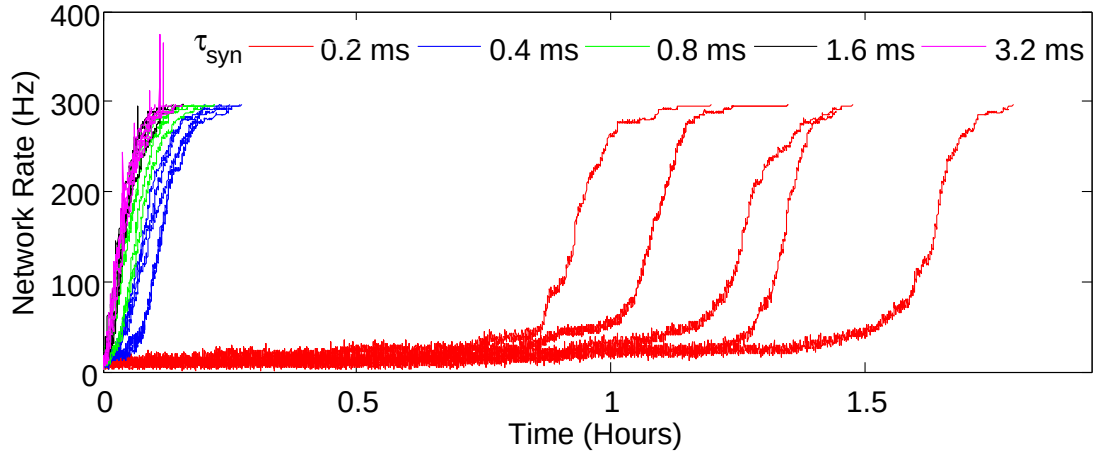


Figure 4.13: Network rate of simulations with increasing synaptic time constants. A short time constant gives a slower development rate.

time constants. However, chains will destabilise if the time constant is too long (greater than 3.2 ms). The vulnerability of chain development to multiple successive spikes is a weakness of the model, which could possibly be addressed by reducing learning rates by orders of magnitude or including inhibition in the network. Clearly, biological networks are robust to isolated spikes and background noise. Replicating this robustness is an important topic for future work.

4.2.2.2 Variable Delay

In this work so far, the delay between neurons has been homogeneous for a given network. Such uniformity is not biologically realistic; the variability in neural morphologies imparts a range of axonal and dendritic delays, in addition the stochastic nature of synaptic transmission ensures diversity in transmission times. In previous works using classical STDP, distributions of delays have produced braided structures or synfire braids [13,73,75]. Here, the robustness of synfire chain development with triphasic STDP is addressed. Specifically, what level of delay variability can chain development withstand and whether there is a identifiable transition point to braids. To answer these questions, simulations were run with the delay between network neurons drawn from a normal distribution. The mean of the distribution was fixed at d (5 ms) whilst the standard deviation, σ_d, was modified from 0.1 ms to 1.0 ms.

Figure 4.14 displays the final activity of the networks. Unlike previous work [13,

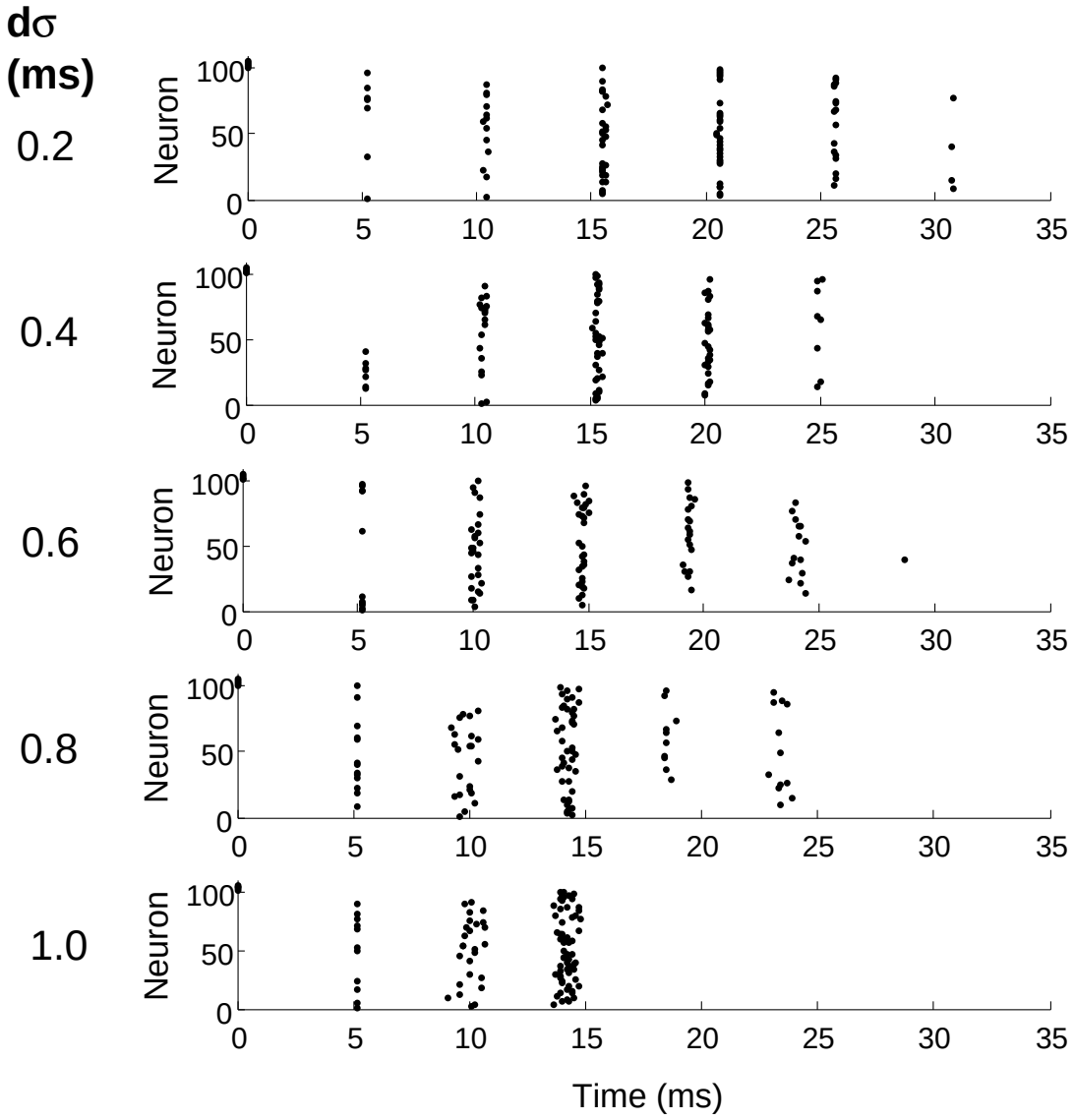


Figure 4.14: Final activity of networks with heterogeneous delays. Increasing the range of transmission times induces jitter within layers and reduces the number of layers.

73, 75] synfire braids are not observed. Instead, as before, chains of activity are observed with feed-forward transmission through a series of layers. However, including variation in the delay between neurons alters both the synchronised firing previously observed within layers and the number of layers that develop. In the previous results each layer fired synchronously exactly d ms after the previous layer (or input). Here, there is significant variation in spike times within the layers, which increases with increased variation in the delay times. Those neurons with shorter transmission delays from the previous layer will spike first and the those with longest transmission delays will spike last.

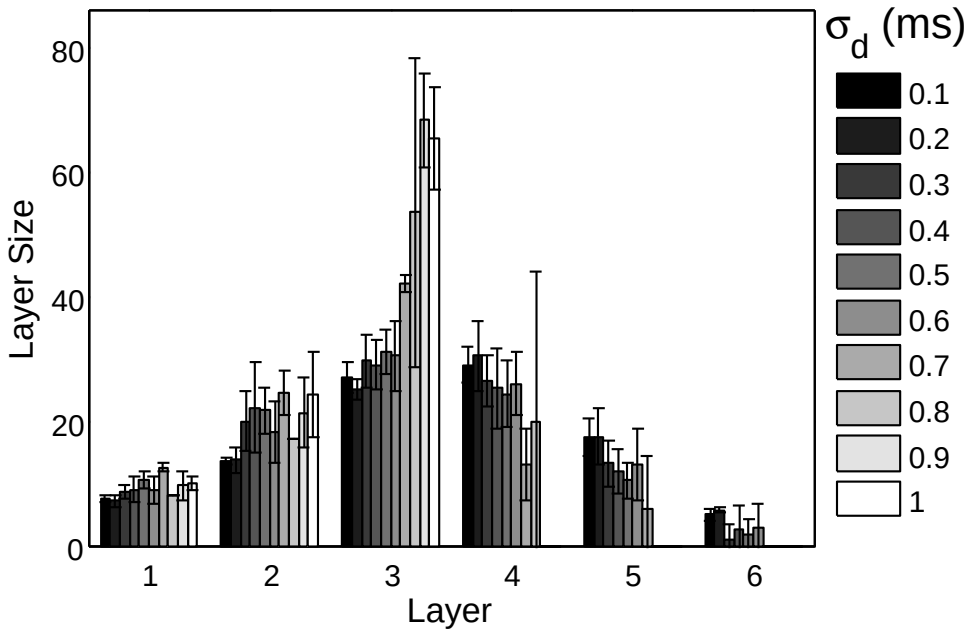


Figure 4.15: Final network structure when delays are drawn from a normal distribution. Larger variation in delays reduces the number of layers.

When the transmission delays are taken from a normal distribution with relatively small standard deviation ($\sigma_d \leq 0.6$ ms), the final structure of the chain is not significantly altered (Figure 4.15). However, when the standard deviation is large ($\sigma_d \geq 0.7$ ms), the resulting chain is reduced in length having fewer larger layers. The explanation for this reduction in layer size is analogous to the collapse problem described earlier for classical STDP (Section 2.1.3.6 and Figure 2.21). There, long range projections were potentiated through the exponential tail of the classical STDP rule and the layer structure 'collapsed' into synchronous firing. Here, the triphasic STDP rule is positive when the spike-time difference is less than 8 ms. Hence, when two sub 4 ms delays occur consecutively the spike-time difference between neurons two layers apart will fall into the positive region of the STDP function and the connection between them will be potentiated, as a result the second neuron will become a member of layer 2 (Figure 4.16).

In conclusion, triphasic STDP can be used to grow synfire chains in networks with heterogeneous delays if the range of delays is not too great. If too many delays fall below the α parameter of the STDP rule (Equation 4.3), neurons will migrate towards the earlier layers, causing a reduction in the final number of layers. However, it is possible to conceive of a situation where this migration of neurons may be a positive ordering mechanism rather than a problem. Where the transmission between neurons is very small, it might be the case that they are actually part of the same functional group and such a

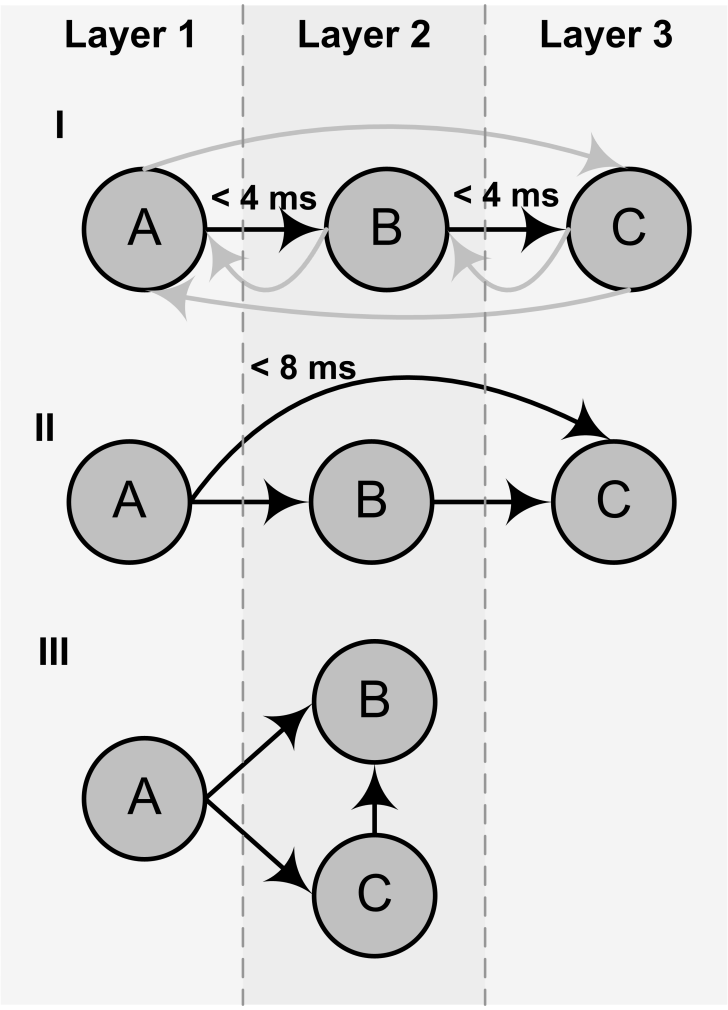


Figure 4.16: Including delays that are less than 4 ms will cause some neurons to migrate to an earlier layer.

function would help synchronise related neurons.

Finally, this section raises some interesting questions for future research which there was not time to address here. Firstly, would a sparse network (biological networks are reported to have connectivity at around 6 % [113]) reduce the risk of sub-chains with sequential short delays and hence be able to support a range of delays without collapse as in [73, 75] possibly leading to braid structures. Secondly, there is huge variation in the triphasic STDP rules reported [32, 116, 167]. How would a network with triphasic STDP using heterogeneous α values develop? The result could resemble [13] where the plasticity rule was bolted to the specific transmission delay in question. Synfire braids were also reported here.

Table 4.3: Model parameters for large networks.

	Simulation Parameters
N	10,000
N_{in}	200
d	5.0 ms
λ_p	0.1 Hz
λ_{in}	3 Hz
W_{max}	20 nS
A	8
τ_{syn}	0.2 ms
V_{th}	-50 mV
t_{ref}	20 ms
E_L	-85 mV
V_{reset}	-80 mV
V_m	-80 mV
E_{ex}	0 mV
g_L	1.125 nS
C_m	22.5 pF

4.2.2.3 Realistic Network Sizes

The final part of this chapter addresses the question raised in [86]. Here, the authors show that synfire chain development through classical STDP is unstable with synfire chains either failing to propagate or alternatively recruiting the entire network. The determinant being the size of the input group. They suggest that as all previous successful attempts to grow chains have been a result of either very small networks or very low numbers of synaptic connections the results are "artifactual". They suggest that a realistic network model should contain tens of thousands of neurons with low (6%) connectivity. The results in this thesis so far are also on small networks. Therefore, here the size of the network is increased considerably to 10,000 with random connectivity at 6%. To account for such an increase in the scale of the simulations, other neural parameters have to be adjusted. The parameters for the following simulation are given in Table 4.3.

The final network structure is presented in Figure 4.17. The mechanism scales to realistic network sizes. 8551 of the 10,000 neurons were recruited into a single chain of eleven layers. The simulation was run for the maximum time permitted on the University of Leeds high performance computing facility (48 hours), although, this was less than five minutes of simulated time. The combination of triphasic STDP and activity-dependent excitability scales up to realistic network sizes. The reduced connectivity increase the number of layers, whilst maintaining the diamond like shape. In a network of this size, it

should be possible to embed multiple chains. To do this, one could include multiple input groups as in Section 3.3.7

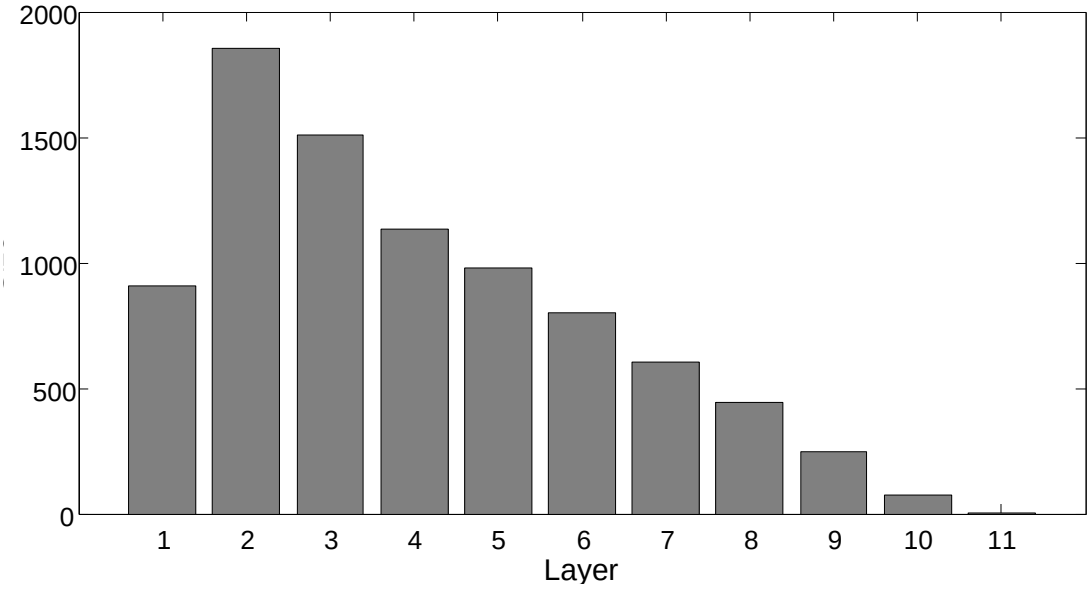


Figure 4.17: Structure of network with 10,000 neurons, all other parameters as Table 4.3. Note only 8551 of the neurons were recruited due to limits on computational time.

4.3 Conclusions

In this chapter triphasic STDP was combined with an LIF neural model to study the development of synfire chains in neural networks. Validating the results of the previous chapter using a more realistic neural model demonstrates that the previous findings were not dependent on the simple binary model previously used. In this conductance based leaky integrate and fire neural model, the neurons are able to aggregate their inputs over time scales in the tens of milliseconds. As a result, transmission between layers can occur without synchronous firing at individual layers. Thus, variability in transmission delay or synaptic time scales can be incorporated into the model. Development of synfire chains through triphasic STDP was found to be robust to this change of neural model and as such the developmental mechanism does not depend on any particular model. In addition, it was shown that triphasic STDP plus activity dependent plasticity can be used to grow synfire chains in a network of realistic size and connectivity.

Here activity dependent excitability was modelled as sparse spontaneous activity that terminated once the neuron was part of an active chain (defined as firing within 1 Hz of the

input). Activity dependent excitability is well reported [109, 150, 154, 165], however, this implementation could be considered over simplistic. It would be possible to implement such excitability using a specifically designed a neural model, which either through a relative refractory period or transient calcium current displayed the required behaviour. By modelling the mechanism of the activity dependent excitability, such a model would indeed add an element of biological realism. However, given that it would not effect the neurons' spike-times it is conjectured that such a modification would not qualitatively affect the results.

In addition, the model presented here allowed investigation into the effect of heterogeneous network delays. It was shown that the model is robust to slight variations in transmission delay. However, if the variation is too large, the network neurons can be seen to migrate to earlier layers. This poses an interesting question: could such behaviour, categorised here as undesirable, have a function in developing networks? For example, if we know that X always gives Y and Y always gives Z, it may be useful to infer that the relation is transitive and X gives Z, thus this layer migration is a form of learning in our network. Understanding what structures would result from either employing a range of triphasic STDP functions or non-normal delay distributions are important open questions for further work. It may be the case that other structures such as braids or even stable recurrent structures could be developed. Were this the case, it would lead to yet another question. How does the brain determine which rule to implement and when?

It has been shown that triphasic STDP can be used to develop synfire chains in networks of binary neurons or networks of LIF neurons of various sizes. However, recall that the function used in this and the previous chapter is simply one of a class of functions that could suit our purpose, which was chosen because of its biological realism. In the next chapter an attempt is made to understand the developmental process even more deeply by removing the neural model entirely. There we ask: what is the simplest model that describes synfire chain development?

Chapter 5

Growing Synfire Chains: A Random Walk Description

In the previous chapters the development of synfire chains has been investigated using computational models which explicitly model spike times and the corresponding plasticity. A characteristic diamond-shaped network structure was observed throughout these investigations. A network's profile (i.e., the relative sizes of individual layers) is determined by the network size, the input size and the learning rate. Here a description of network growth is developed which predicts the final network shape from the initial parameter regime. The stochastic nature of the spontaneous firing allows one to describe the plasticity events as steps on a random walk. This formulation is used to develop a set of equations which describe network development; thus investigation into the possible range of network shapes can be done quickly and without neural simulation. The chapter begins with an introduction to random walks before detailing their applicability to the synfire chain development problem. The initial application is to determine the first recruitment time; this is then extended to determine the entire development trajectory. Results are compared to the neural simulations of Chapter 3, which exposes hitherto unreported aspects of the process. Simplifications to these simulations are then presented which align the predictions of the random walk description and the simulation results. Finally, this simple description is used to investigate the importance of the balance between potentiation and depression in the triphasic STDP rule.

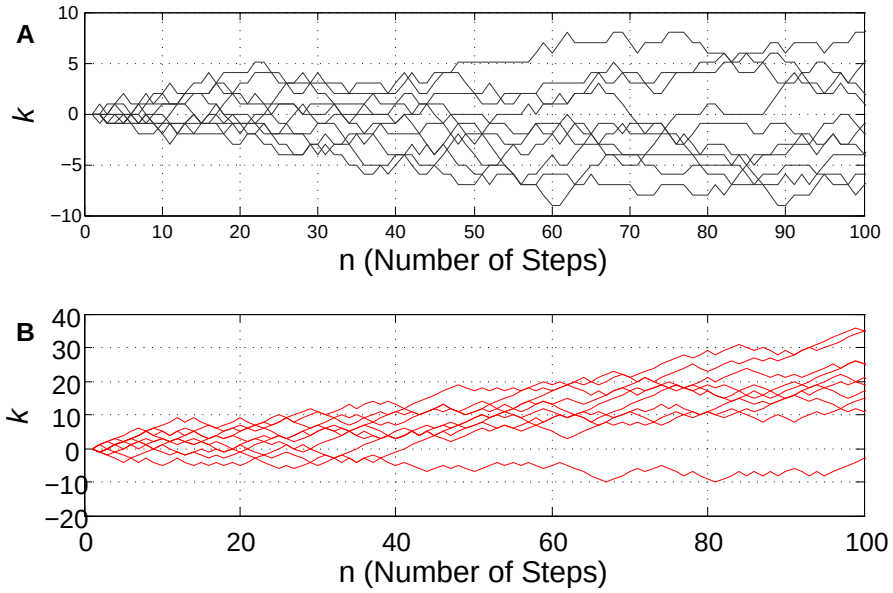


Figure 5.1: Random walk examples. Ten instances of a random walk with p = q = 0.5 (A) and with p = 0.75 and q = 0.25 (B).

5.1 Random Walks

The meanderings of a beetle, the migration of cells, the increase and decrease in a penguin population's size. All of these biological processes and many more can be described as a random walk. In a 1D random walk (Figure 5.1) there is a probability of stepping down (q) or up (r). The probability of being in position k after n steps is defined here as $p_n(k)$ and is described by the recurrence relation,

$$p_n(k) = qp_{n-1}(k-1) + rp_{n-1}(k+1). \tag{5.1}$$

If r equals q the walk is unbiased Figure (5.1 A). Any bias ($r \neq q$) leads to an expected drift in position k (Figure 5.1 B).

In Equation (5.1) k can take any value in the range $\{-\infty, +\infty\}$, but imagine our meandering beetle is on a window ledge, with the window on one side and a long fall on the other. Now we have a bounded random walk. The window is a reflecting boundary as the beetle will simply bounce back onto the ledge. The edge of the ledge is an absorbing boundary as the beetle falls to its fate it will not return to the random walk. The average time taken to hit such a boundary is termed the mean first passage time (MFPT). (See Refs. [45, 158] for thorough texts on random walks).

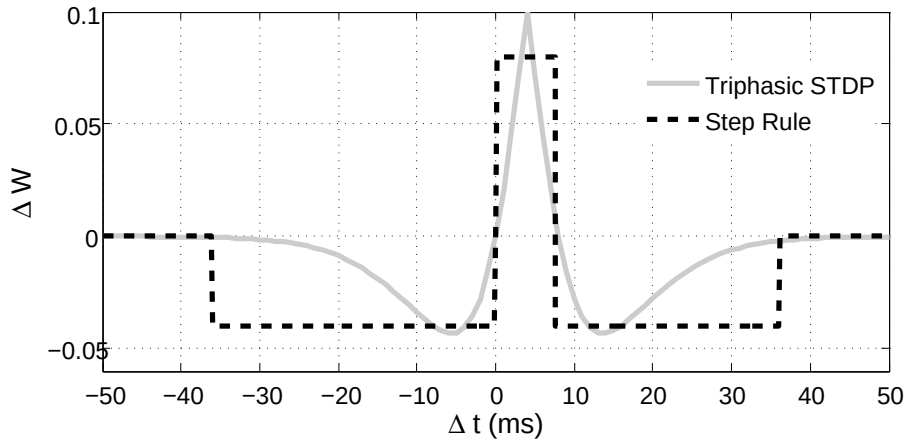


Figure 5.2: Triphasic STDP (grey) with α=4 ms and A=0.1. Step function (black dashed) with A_d=-0.04, A_p=0.08, τ_{d-}=-36 ms, τ_p=7.5 ms and τ_{d+}=36 ms

5.2 Random Walks and Synfire Chain Development

The development of synfire chains as observed in the preceding chapters can be described as a collection of random walks. Each synapse in the network undergoes its own random walk. There are $N(N+N_{\text{in}})$ synapses, therefore, $N(N+N_{\text{in}})$ parallel random walks. The steps of the walk are analogous to spontaneous spiking events, which occur randomly at a fixed rate for each neuron. Each walk is one dimensional with two boundary conditions: a reflecting boundary at zero (the weights cannot change from excitatory to inhibitory) and an absorbing boundary at the synaptic weight at which recruitment occurs, w_R. A key assumption throughout this chapter is that once a neuron is recruited, it remains in its fixed position within the chain. This is a valid assumption within the successful parameter regimes outlined in the preceding chapters. A consequence of this assumption is that the random walk is a description of development until the point of recruitment. The stability of the chain whether established through triphasic STDP or some other means is not addressed by this description.

The triphasic STDP function in Equation (3.4) is a smooth function that maps spike-time differences to plasticity, giving a range of different potentiation and depression steps. For tractability, here the function is reduced to a simpler piecewise constant function (henceforth dubbed a "step rule") given by Equation (5.2) and illustrated in Figure 5.2. This function approximately replicates the overall ratios of potentiation and depression from the triphasic rule (Equation (3.4)). As in the triphasic rule, depression occurs over a much wider range, but with lower magnitude, and potentiation is limited to spike time

differences that are less than two transmission delays.

$$\Delta W_{ij} = \begin{cases} A_d & \tau_{d-} < \Delta t < 0 \\ A_p & 0 < \Delta t < \tau_p \\ A_d & \tau_{p+} \leq \Delta t < \tau_{d+} \\ 0 & \Delta t \leq \tau_{d-}, \Delta t \geq \tau_{d+}, \Delta t = 0, \end{cases} \tag{5.2}$$

A_d is the depression step, A_p the potentiation step, τ_p the cross over from potentiation to depression, τ_{d-} and τ_{d+} the limits of the depressive area.

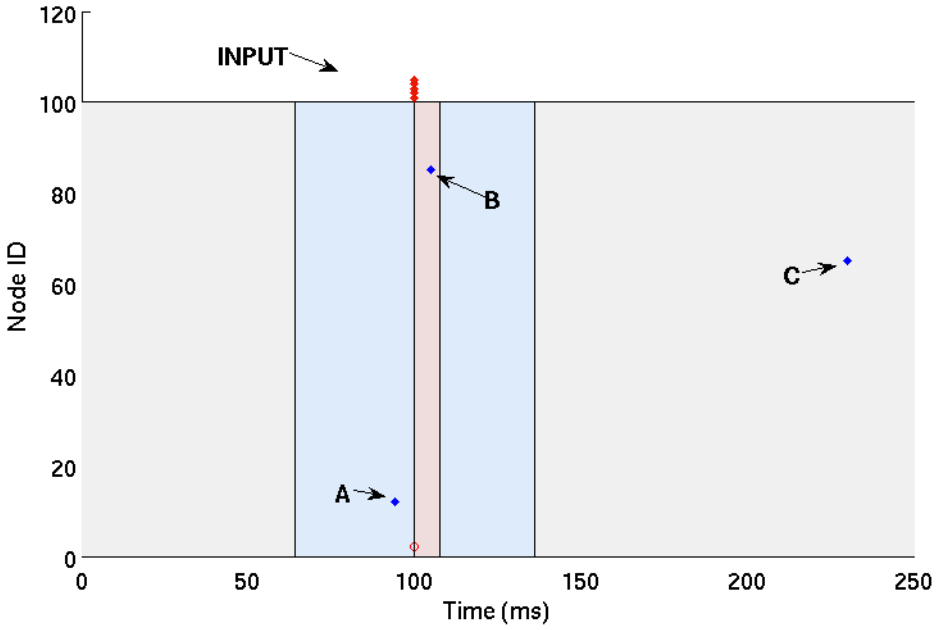


Figure 5.3: Pictorial representation of the probability of making either a positive or negative step. The probability of a potentiation event within one presentation of the input equals the probability that a spontaneous spike occurs within a τ_p (7.5 ms) window following the input, shaded pink. Similarly for depression, except here two longer blue windows flank the potentiation window. Spike A results in depression, B potentiation, C no change.

When using the simplified STDP rule, the steps in the random walk are of fixed magnitude. To calculate the probability of a plasticity event occurring it is convenient to assume the spontaneous firing follows a Poisson process. This is not strictly correct as the refractory period of the neurons prevents two spikes occurring in quick succession ($< t_{\text{ref}}$). However, because the spontaneous rate is very low ($\lambda_p << 1$) this small difference is negligible. Similarly, it is convenient assume that a plasticity event can happen at any time; the random walk is described in continuous time. This neglects the regularity of

the input neurons' firing. Given these approximations, it is now possible to estimate the probabilities per second of making a potentiation step, p, or depression step, d, from the spontaneous rate, λ_p., the input rate, λ_{in}, and the τ_p, τ_{d-} and τ_{d+} parameters of the step rule (Figure 5.3) [45].

The probability per second for the potentiation of a given synapse, p, is constant and given by

$$p \equiv \lambda_{\text{in}} \lambda_p \tau_+ e^{-\lambda_p \tau_p}. \tag{5.3}$$

For depression a similar formula holds,

$$d \equiv \begin{cases} 0 & W = 0 \\ \lambda_{\text{in}} \lambda_p \tau_d e^{-\lambda_p \tau_d} & W > 0, \end{cases} \tag{5.4}$$

where τ_d is the combined width of both depression windows ($\tau_d = \tau_{d+} - \tau_{d-} - \tau_p$) and synaptic depression is limited to positive weights.

5.2.0.4 First Recruitment Time

As observed in the previous chapters, different layers within the developing network recruit neurons at different rates and these layers compete with each other. To reduce the level of complexity in the random walk description, a simpler initial problem is identified. In the initial stages of development, before any neurons are recruited, each neuron can be considered independent and undergoing a random walk with the input. This scenario is considered first, with specific attention given to the expected recruitment time. For the step rule, the probability of potentiation or depression of a given synapse is constant and given above. If we take A_p to be double A_d the possible set of weights a synapse can have is discrete (Figure 5.4) and defined as

$$W = nA_d,\ n \in \{0, 1 \ldots R-1\}, \tag{5.5}$$

where R is the point at which a synapse leaves the walk. Thus, up until recruitment time, a pool neuron's synaptic weights hop among each of the R bins in a random walk, with the requirement for positive (excitatory weights) represented as a reflecting boundary condition at $n = 0$. Since each neuron sees identical input spikes, the input weights to any given neuron move in tandem. Therefore, the weight trajectories of all synapses from the input neurons to a single pool neuron can be modelled as a single random walk. Initially input spikes will not cause postsynaptic firing. Only when the combined input reaches the firing threshold $N_{\text{in}} W \geq \theta$ will a postsynaptic spike be generated, immediately

recruiting this neuron to the first layer. This synapse leaves the pool and its random walk is terminated. This is represented by an absorbing boundary condition at $R = \lceil \theta/(N_{\text{in}}A_d) \rceil$, so the cut off weight for recruitment is $W_R = \theta/N_{\text{in}}$. The random walk is only defined between 0 and W_{R-1}, after this point the behaviour is assumed. To find an estimate for the time a synapse leaves the random walk (is recruited), its trajectory before this point must be modelled.

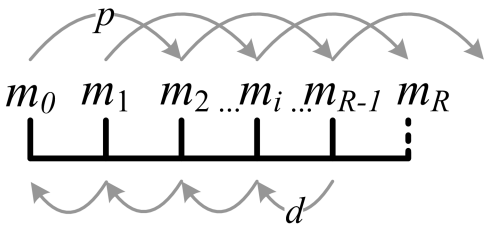


Figure 5.4: When the step rule is used, the synaptic weights fall into one of a set of discrete bins. Synaptic weights hop among the bins with probability per unit time p and d. If a weight reaches the final bin, R, the neuron is recruited and the weight exits the random walk.

Assuming very large N or many repeats of the experiment, the expected value for any given synaptic weight at a particular time, t, is determined by the hops previously made in the random walk and is defined as a probability mass function,

$$\vec{m}(t) = \langle m_0, m_1, \ldots, m_{R-1} \rangle. \tag{5.6}$$

$\vec{m}(t)$ has R elements, each corresponds to a possible synaptic weight and is identified by the bin index $n \in \{0, 1 \ldots R-1\}$ where m_n is the probability of an efferent synapse occupying bin n, i.e., having a weight which equals nA_d. The elements of $\vec{m}$ are non-negative and sum to a value equal to or less than 1. It is now possible to approximate the stochastic process of recruitment to the first layer by a master equation for $\vec{m}(t)$ [45, 87, 158]

$$\frac{d\vec{m}}{dt} = T \cdot \vec{m}, \tag{5.7}$$

the transition matrix T for all cases where A_p is twice A_d can be summarised by

$$\begin{aligned} \frac{dm_n}{dt} &= p\,m_{n-2} + d\,m_{n+1} - (d+p)\,m_n \qquad 1 < n < R-1\,, \\ \text{with boundary conditions} & \\ \frac{dm_0}{dt} &= d\,m_1 - p\,m_0 \\ \frac{dm_1}{dt} &= d\,m_2 - (d+p)\,m_1 \\ \frac{dm_{R-1}}{dt} &= p\,m_{R-3} - (d+p)\,m_{R-1}\,. \end{aligned}$$

and initial conditions (full occupancy at m_0),

$$\vec{m}(0) = \langle 1, 0, \ldots, 0 \rangle. \tag{5.8}$$

The transition matrix T is an asymmetric tri-diagonal Toeplitz matrix which will be different for different parameter regimes. In the regime previously described ($A_d = 0.4; A_p = 0.8; \theta = 1; N_{\text{in}} = 5$), $R = \lceil \theta/(N_{\text{in}} A_d) \rceil = 5$ and T is given by,

$$T = \begin{pmatrix} -p & d & 0 & 0 & 0 \\ 0 & -(p+d) & d & 0 & 0 \\ p & 0 & -(p+d) & d & 0 \\ 0 & p & 0 & -(p+d) & d \\ 0 & 0 & p & 0 & -(p+d) \end{pmatrix}. \tag{5.9}$$

Initially all synapses are weight 0, hence $m_0 = 1$ and all other bins are empty. However, once sufficient plasticity steps have occurred, the occupancy of a mid-range bin n (where $1 < n < R-1$) will increase through potentiation from bin $n-2$ at rate p, or through depression from bin $n+1$ at rate d and decrease through either potentiation or depression at a rate $d+p$ (diagonal and sub-diagonal terms in Equation (5.9)). Due to the reflecting boundary condition, bin 0's occupancy is governed only by depression onto it and potentiation from it. The upper absorbing boundary condition eliminates depression from bin R to bin $R-1$.

It is now possible to estimate the mean first passage time, which gives the expected time of the first recruitment. First, calculate the probability at time t that recruitment hasn't yet happened. This is given by the entire contents of bins $0, \ldots, R-1$

$$\sum_{n=0}^{R-1} w_n(t)\,. \tag{5.10}$$

Conversely, the probability of recruitment is given by that which has left the system. Hence, (following [158]) the probability density for single synapse hitting the recruitment weight, W_R, $f_i(t)$, is given by,

$$\begin{aligned} f_i(t) &= -\frac{d}{dt}\sum_{n=0}^{R-1} w_n(t) \\ &= pw_{R-1}(t) + pw_{R-2}(t)\,. \end{aligned} \tag{5.11}$$

The expected recruitment time (mean first passage time) $\bar{t}_i$ is hence given by

$$\bar{t}_i = \int_0^\infty t f_i(t)\, dt\,. \tag{5.12}$$

Having found $\bar{t}_i$, it is now possible to estimate the time of the first recruitment in a network with N pool neurons. To find the probability that at time t at least one neuron has been recruited, $f_N(t)$, calculate the complement of the probability that none have been recruited between 0 and t,

$$\begin{aligned} f_N(t) &= 1 - \left(1 - \int_o^t f_i(t) dt\right)^N \\ &= 1 - \left(\sum_{n=0}^{R-1} w_n(t)\right)^N\,. \end{aligned} \tag{5.13}$$

The expected first recruitment time is then given by $\bar{t}_N$,

$$\bar{t}_N = \int_0^\infty t f_N(t)\, dt\,. \tag{5.14}$$

Solutions to Equation (5.14) can be solved analytically for small N and are given by numerically solving the master equation (5.7). The solution of $\vec{w}$ over time (Figure 5.5) shows that the recruitment process onto the first layer can be described as a rapid escape (or multiple very closely spaced potentiation events) from the unrecruited pool. Initially all weights are zero: All probability is concentrated in bin $w_0(0) = 1$. This bin initially depletes rapidly due to potentiation and some bins will have higher occupancy, i.e., $w_{1,\ldots,R-1}$ will begin to grow. As these bins are populated, depression comes into play, reducing most weights to 0 again. The net result is that the probability in higher bins remains almost stationary at a very low value while the first bin, w_0, depletes very slowly (due to recruitment).

In a relatively small sized network this means that at any time very few potentiated

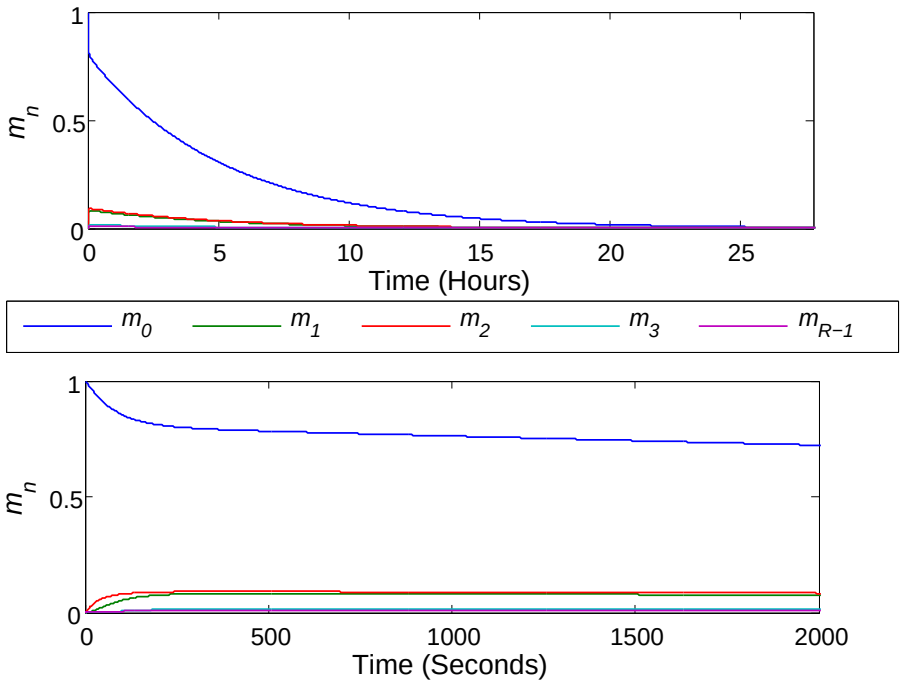


Figure 5.5: The depletion of the bins m_n. After a day of simulated time all weights have been absorbed at the boundary condition m_R (top). At this timescale the depletion due to recruitment by quick escape is dominant. The initial phase (bottom). At this timescale potentiation dominates, quickly populating the higher bins, before a steady state is reached at approximately 200 seconds

weights are visible: recruitment into the first layer takes place if a neuron's weights happen to potentiated rapidly a number of times in relatively close succession. This is indeed what was observed in the neural simulations (Figure 3.12). With this analysis it is now possible to estimate the time at which the first layer starts to form directly in terms of the simulation parameters (Figure 5.6).

To validate the estimated the time for the first recruitment, a simulated random walk was constructed. A vector of size N is created and initialised with zeros. At each time step (1 second) each element of the vector is increased by A_p with probability p and decreased by A_d with probability d (unless this causes the value to fall below zero, when it is set to zero). The time is recorded when the first of the elements reaches W_R. To get a estimated mean recruitment time the walk is repeated 10,000 times. Figure 5.6 shows that this simple random walk matches the analytic result and is a good estimate of the first recruitment times seen in the binary simulations.

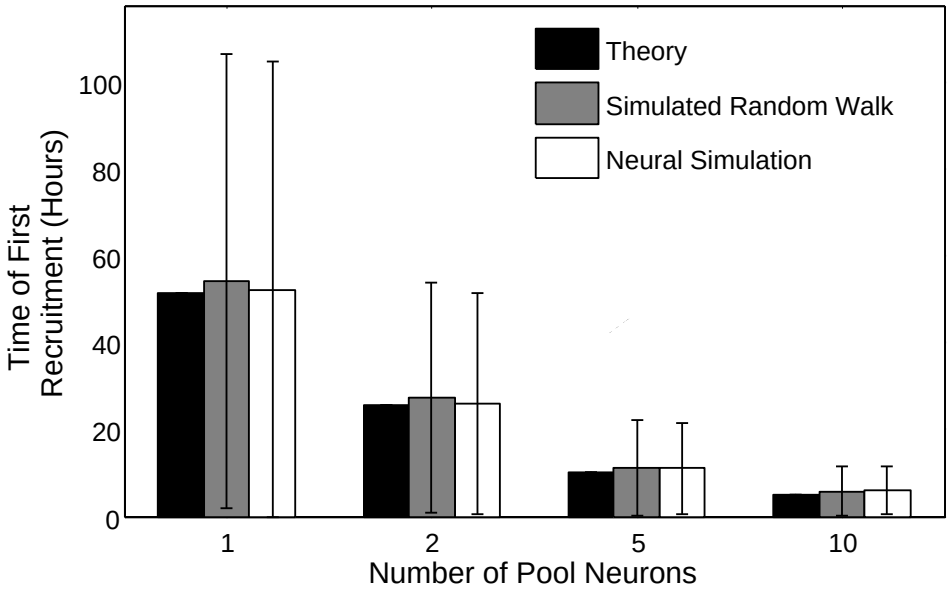


Figure 5.6: The expected time of the first recruitment as calculated compared to the mean and standard deviation of first recruitment times in random walk simulations (10,000 runs; $A_p = 0.08$; $A_d = 0.04$; $\theta = 1$; $\lambda_p = 0.1Hz$; $\lambda_{\text{in}} = 3Hz$; $\tau_p = 7$ms; $\tau_{\text{d}-} = -36$ms; $\tau_{\text{d}+} = 36$ms; $\text{N}_{\text{in}} = 5$) and neural simulations (1000 runs, using the step rule and parameters as above).

5.2.0.5 Subsequent Layer Formation

When describing the development of the entire chain, think now not of one random walk but of multiple concurrent random walks. If plasticity events were independent, these random walks would exactly describe chain development in the system. In fact, the spike time dependence in the learning rule introduces some correlations in plasticity events. Here, the random walk formulation is described for multiple layers. Then the adjustments to the neural simulations are described, which remove such correlations as far as is possible. Finally, the random walk formulation is compared to both the simplified and the full neural simulations.

Previously, in the random walk model of the first recruitment, the threshold, R, at which recruitment occurred was fixed and determined by the input size, N_{in} and the firing threshold, θ. However, layer sizes within the chain are not fixed and as a result R becomes a dynamic variable which depends on the recruiting layer size, N_ℓ. In addition, the synapses from a layer onto a particular pool neuron may have formed at different times and thus a layer's weights to that pool neuron are heterogeneous. It is helpful therefore

to model the total efferent weight onto a pool neuron taking these dynamics into account. Thus recruitment occurs when the combined efferent weight reaches the firing threshold, $\sum_{i=1}^{N_\ell} W_{ij} \geq \theta$. For every pool neuron, the combined afferent weight $W_{\ell i}$ undergoes a random walk with dynamic step sizes for potentiation $A_p \cdot N_\ell$ and depression $A_d \cdot N_\ell$.

These random walks were simulated explicitly. Since all weights from any layer to a any pool neuron undergo plasticity in sync, it suffices to simulate L random walks for each pool neuron, where L is the (dynamic) number of layers in the system. Each weight $W_{\ell i}$ is defined as total efferent weight from the layer $\ell \in \{0, \ldots, L\}$ to a neuron, i, initially set to 0. The magnitude of the plasticity steps are $A_p N_\ell$ (for potentiation) and $A_d N_\ell$ (for depression), where N_ℓ is the size of the recruiting layer (or the input in the first instance). As before, probabilities to potentiate and depress are given by p and d, respectively. Starting from $N_0 = N_{\text{in}}$ and $N_{\ell>0} = 0$, all random walks are run concurrently. Each recruitment event ($W_{\ell i} > \theta$) leads to an increment in the corresponding layer size $N_{\ell+1}$. Once a neuron has been recruited, all random walks associated with that pool neuron are halted. The short script that executes the random walk is included on the accompanying CD.

The network shape predicted by the multilayered random walks is presented in Figure 5.7 with the results from the binary simulations of Chapter 3. Note a slight difference in the results. To align the results from the two models simplifications need to be made to the neural simulator. Up to these simplifications the random walk description gives a complete description of the neural development process.

The first simplification to the simulations described in Chapter 3 is to use the simpler "step rule" which has strictly zero plasticity for spike time differences greater than $\pm$36ms). Further differences are the result of network interactions not accounted for in the random walk description. Consider three classes of random walks. Firstly, the recruitment of the initial layer is driven by plasticity events between the pool neurons and the inputs, this is fully described by the simple random walk in Section 5.2.0.4.

Secondly, there is also plasticity and therefore random walks between the individual pool neurons. As the spontaneous rate is low, the intra-pool interactions occur with much lower probabilities than those involving either inputs or recruited neurons. Nonetheless at the time of recruitment onto the chain, a pool neuron may already have some non-zero weights in its connections with the remainder of the pool. To neglect the effect of the second set of (intra-pool) random walks, the first simplification made to the simulation protocol described in Chapter 3 is resetting efferent weights of newly recruited neurons to 0. Simulations including intra-pool dynamics give quantitatively similar results, indicating that the contribution of intra-pool dynamics is very small (Figure 5.7).

Finally, during recruitment to subsequent layers of the chain, each pool neuron un-

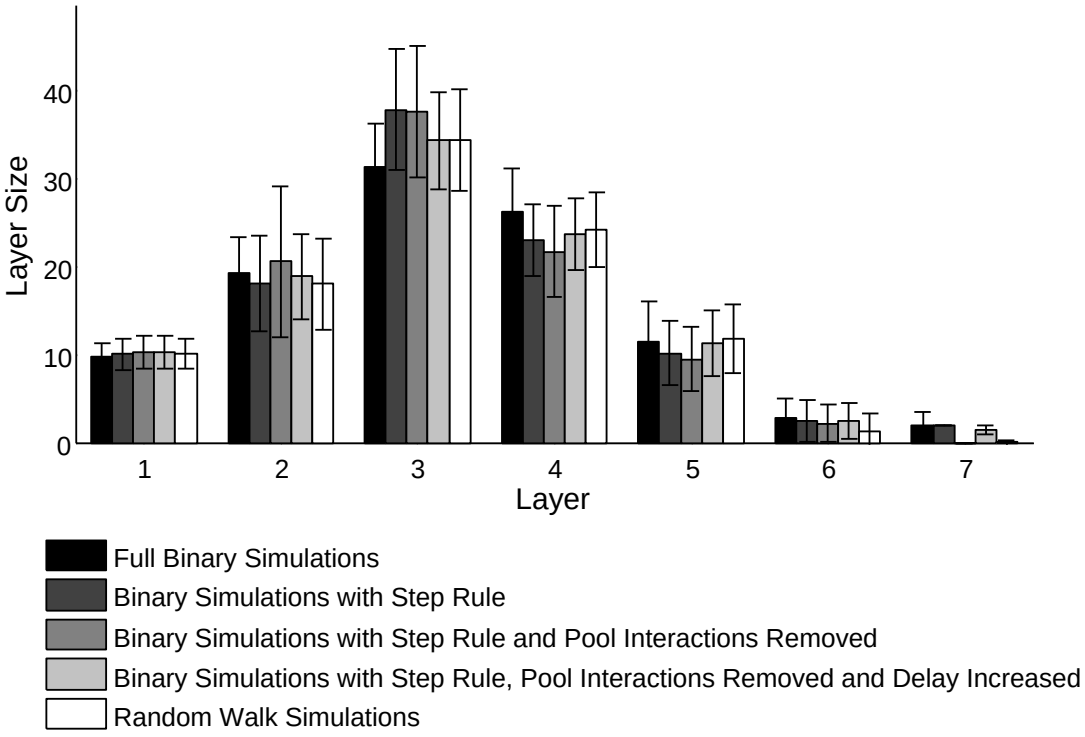


Figure 5.7: Multi-layer chain structures that evolve from the neural simulations versus random walk simulations (white). Mean and standard deviation of layer sizes were calculated over 500 runs. See text for full simplification details.

dergoes random walks that compete to recruit it onto each of the existing layers in the chain, this is described by the multilayered walk detailed above. In the neural simulation the competition to recruit is made more complex by overlapping potentiation windows. Each layer (and the input) will recruit neurons that emit the required number of spikes within the potentiation window of τ_p duration following each layers activation. However, in the previous chapters, the delay (d) has been less than τ_p (typically 5 ms and 8 ms respectively). As a result, the potentiation windows of two consecutive layers overlap (typically by 3 ms). When a spike occurs within this overlap the layers compete to recruit the neuron. In the simplest case, when the plasticity is through the step rule and the layers are large (R=1), the neuron is initially recruited by both layers. On the next presentation of the input, the neuron fires following the earlier layer and is in refractory when the incoming spikes are received from the later layer. The neuron does not fire again, and the potentiated synapses from the later layer now depress. When the smooth triphasic rule is used or when $R > 1$ the situation is more complex. To remove this complexity, the neural

simulations were run with d set to τ_p - 0.1 ms thus eliminating the overlap.

The near match even to the full neural simulations shows that for the parameter regimes described here, the recruitment dynamics is well described by a set of independent recruitment events. Furthermore, the tractability of the random walk description allows insight into the recruitment process that may not be immediately evident from the simulations, most notably, the relationship between layer size and recruitment rate.

5.3 Development as a Recurrence Relation

In Section 5.2.0.4 it was shown that the recruitment of a single node can be described by the master equation (5.7). Here this simple description is extended to describe the recruitment probability for all layers. A recurrence relation gives the expected network shape for any parameter set. Note that as before, the stability of the chain is assumed and as a result this formulation can only tell us what the chain will look like *if* development is successful.

A probability mass function must be defined for each layer ℓ, $\vec{m}^\ell(t)$, where as before the probability of being in bin n is given by $\langle m_0^\ell, m_1^\ell, \ldots, m_n^\ell, \ldots, m_{R^\ell-1}^\ell \rangle$. Now, however, as the number of neurons in each layer changes with time, so does the value of R^ℓ and consequently the transmission matrix, T. The number of possible bins, R^ℓ, depends on the relative size of potentiation (A_p) and depression steps (A_d). If, as before, one is an integer multiple of the other, then the smallest defines the smallest hop size a. If this is not the case, then the smallest hop size a is the greatest common divisor[1] of A_d and A_p. For example, if $A_d = 0.3$ and $A_p = 0.5$, $a = 0.1$. For simplicity A_d and A_p are redefined in hops: a_p is the number of hops in a potentiation step (A_d/a) and similarly a_d is the number of hops in a depression step (A_d/a). (In the example above $A_d = 0.3$, $A_p = 0.5$, $a = 0.1$, $a_d = 3$ and $a_p = 5$).

There are now ℓ master equations defined as,

$$\frac{d\vec{m}^\ell}{dt} = T(t) \cdot \vec{m}^\ell(t) \tag{5.15}$$

where $\vec{m}^\ell$ is the probability mass function of the efferent weights from layer ℓ. As before, the number of elements in each vector is determined by the recruitment boundary. This is

[1] as gcd is only applicable to integers, A_p and A_d are multiplied by 1000, before the gcd is calculated, the result is then divided by 1000.

now dependent on $N_\ell(t)$ and given by,

$$R^\ell(t) = \left\lceil \frac{\theta}{aN_{\ell-1}(t)} \right\rceil \tag{5.16}$$

where $N_{\ell-1}(t)$ is the number of neurons in the preceding layer at time t. The vector evolves according to the transition matrix $T(t)$, which is a function of $N_\ell(t)$ and therefore must itself be reevaluated at each time point, hence is written $T(t)$. $T(t)$ is therefore defined by the equation,

$$\frac{dm_n^\ell}{dt} = p\,m_{n-a_p}^\ell + d\,m_{n+a_d}^\ell - (d+p)\,m_n^\ell, \qquad a_p \le n < R^\ell - a_d\,,$$

with boundary conditions

$$\begin{aligned}
\frac{dm_n^\ell}{dt} &= \sum_{s=1}^{\max(a_d,R)} d\,m_{n+s}^\ell - p\,m_n^\ell, && n = 0, R^\ell > 1\,,\\
&= -p\,m_n^\ell, && n = 0, R^\ell = 1\,,\\
&= d\,m_{n+a_d}^\ell - (d+p)\,m_n^\ell, && 0 < n < a_p, n \le R^\ell - a_d\,,\\
&= -(d+p)\,m_n^\ell, && 0 < n < a_p, n > R^\ell - a_d\,,\\
&= p\,m_{n-a_p}^\ell - (d+p)\,m_n^\ell, && R^\ell - a_d < n < a_p\,,
\end{aligned} \tag{5.17}$$

and the initial condition $N_0 = N_{\text{in}}$ and $N_0 = 0$ for all $\ell > 0$.

As each unrecruited node undertakes walks with every existing layer, each $\vec{m}^\ell$ sums to the total probability mass left in the system, P, which is initially 1, thus the first element of each vector is initialised at 1, $\vec{m}_0^\ell(0) = \langle 1, 0, \ldots, 0\rangle$. The network size is then determined by repeated application of the recurrence relation,

$$\begin{aligned}
N_\ell(t+1) &= N_\ell(t) + \sum_{n=R-a_p}^{R-1} N(t)\,p m_n^\ell(t)\\
P(t+1) &= P(t) - \sum_{\ell=0}^{\ell=L} \sum_{n=R-a_p}^{R-1} p m_n^\ell(t)\\
\vec{m}^\ell(t+1) &= T(t)\vec{m}^\ell(t) \cdot \frac{P(t+1)}{\sum_n m_n^\ell(t)}
\end{aligned} \tag{5.18}$$

The solutions become stable at $P(t) = 0$ where $N_\ell(t)$ gives the expected size of each layer.

The above formulation can be calculated numerically to give the expected structure for any parameter regime. Figure 5.8 displays the results alongside those predicted by

the multilayered random walk and those obtained from the simplified neural simulations. The expected network structure is compared to structures obtained using the simulated random walk and the binary neural simulation. Three representative parameter regimes were identified and tested. The first, Figure 5.8 A uses parameters closely matched to the original simulations. Figure 5.8 B has a much larger number of inputs and depression steps almost equal to potentiation. steps. Figure 5.8 C has larger depressive steps and a smaller number of neurons. Full parameters are given in the figure caption. The results match well for the three parameter regimes tested (see figure caption). The estimated result is very close to the mean simulation result and therefore shows that the recurrence relation is a useful tool for predicting the outcome of simulations. However, the very slight discrepancy suggests that there may be a further refinement which would result in a perfect match.

5.3.1 Ratio of Potentiation to Depression

It has been argued [143] that, in networks with Poisson firing statistics, the ratio of depression to potentiation is an important factor when developing networks through classical STDP, and that the STDP function should be slightly biased towards depression, hence synapses that see uncorrelated spike trains are depressed. In the triphasic STDP investigations described here the total depression is much greater than potentiation (Figure 5.2). With the recurrence relation description it is now possible to estimate how changing the ratio of potentiation to depression changes the network structure without extensive simulation.

First, whilst keeping the size of the potentiation and depression steps equal, $A_p = A_d = 0.02$ and the probability to potentiate p set at 0.3, the probability to depress is altered. The results are displayed in Figure 5.9. When d is small, the network has fewer larger layers. Increasing d increases the number of layers and also moves the largest layer towards the end of the chain. This result is very similar to that seen in Section 3.3.4 where the effect of varying the input size was investigated. Networks formed with strong depression appear like those formed with few input nodes (compare Figure 5.9 and Figure 3.17). The common factor and cause of this similarity is the number of steps it takes to potentiate. When depression is high, potentiation steps are often reversed by depressive steps before the recruitment target is reached. Hence a single synapses will, on average, have made many more potentiation steps before recruitment than would be necessary without depression. Similarly, when the number of inputs is low, many potentiation steps must be made before recruitment. Therefore, the shape of the final network is a result of

both the initial network configuration and the STDP rule.

The relative range of the depression and potentiation windows has a clear impact on the final network structure. But what about the magnitude of the steps? Here probability to depress equals the probability to potentiate ($p = d = 0.015$), A_p and A_d are manipulated. The results are presented in Figure 5.10, interestingly, it is not the $A_p : A_d$ ratio that is important, it is the value of A_p. The rows of Figure 5.10 have fixed A_p and varying A_d and very little change can be seen in the network shape. However, the columns where A_d is fixed and A_p varies show a transition to shorter and wider chains as A_p increases. The transition with A_p could have been predicted from the results detailed so far. Briefly, more potentiation steps lead to longer, narrower chains. However, the static network shape over

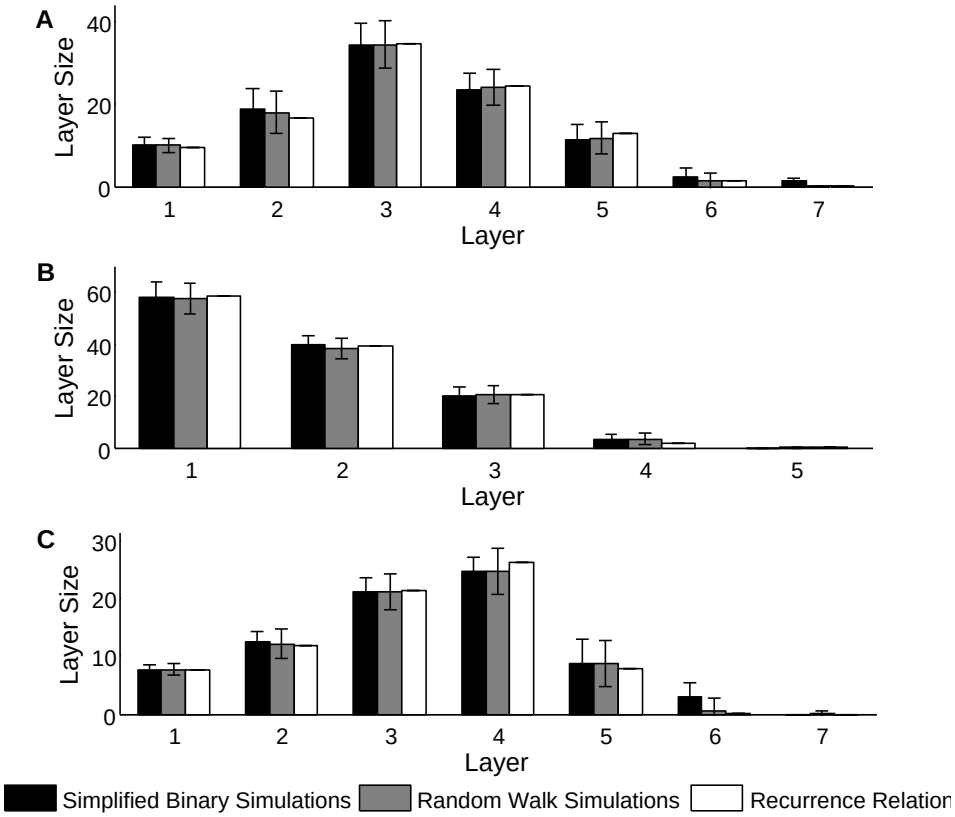


Figure 5.8: Network structures estimated using the three methods. Bars in neural simulation and random walk are mean result for 100 runs with standard deviation as error bars. The three methods give well matched results for **A**: $\theta = 1$, $A_p = 0.08$, $A_d = 0.04$, $N = 100$, $N_{\text{in}} = 5$. **B**: $\theta = 1$, $A_p = 0.05$, $A_d = 0.04$, $N = 120$, $N_{\text{in}} = 20$. **C**: $\theta = 2$, $A_p = 0.1$, $A_d = 0.12$, $N = 75$, $N_{\text{in}} = 5$. ($\tau_p = 7$ms, $\tau_{\text{d}-} = -36$ms, $\tau_{\text{d}+} = 36$ms, $\lambda_{\text{p}} = 0.1$Hz, $\lambda_{\text{in}} = 3$Hz for all.)

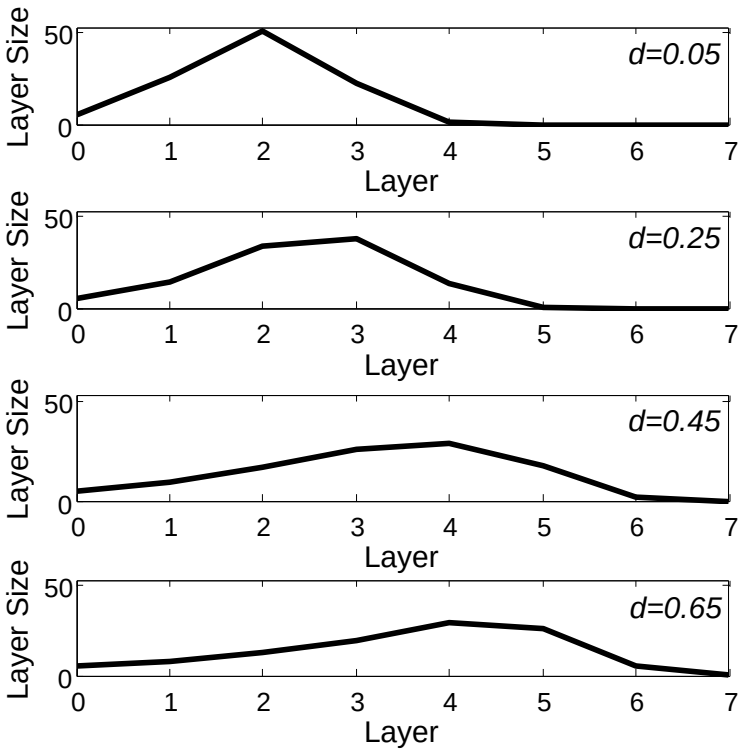


Figure 5.9: The relative probabilities to potentiate (p) or depress (d) alter the network shape. (Results obtained with recurrence relation, Equation (5.18) $\lambda_p = 0.1$Hz, $\lambda_{\text{in}} = 3$Hz, $\theta = 1$, $\text{A}_\text{p} = 0.02$, $\text{A}_\text{d} = 0.02$, $\text{p} = 0.3$, $\text{N} = 100$, $\text{N}_\text{in} = 5$.).

a range of A_d values is unexpected. One might have expected that when A_d was large this would result in more potentiation steps and thus a narrowing and lengthening of the chain. This surprising result is a consequence of the reflecting boundary at zero. As shown previously (Figure 5.5), during development most synaptic weights are at zero with a small fraction in the lower bins and very few in the highest bins. At $w = 0$ the depressive step has no effect. At w_1 the size of A_d is irrelevant, all depressions are bounded at zero. Only when A_p is large does the size of A_d have an effect, when it could take more than one depression step to fall back to zero. This can be observed from Figure 5.10, where in the final row, there is a slight change in the network shape.

5.4 Conclusions

In this chapter synfire chain development was described as a random walk. Developing this simple formulation was instructive not only because of the useful outcome: a way to

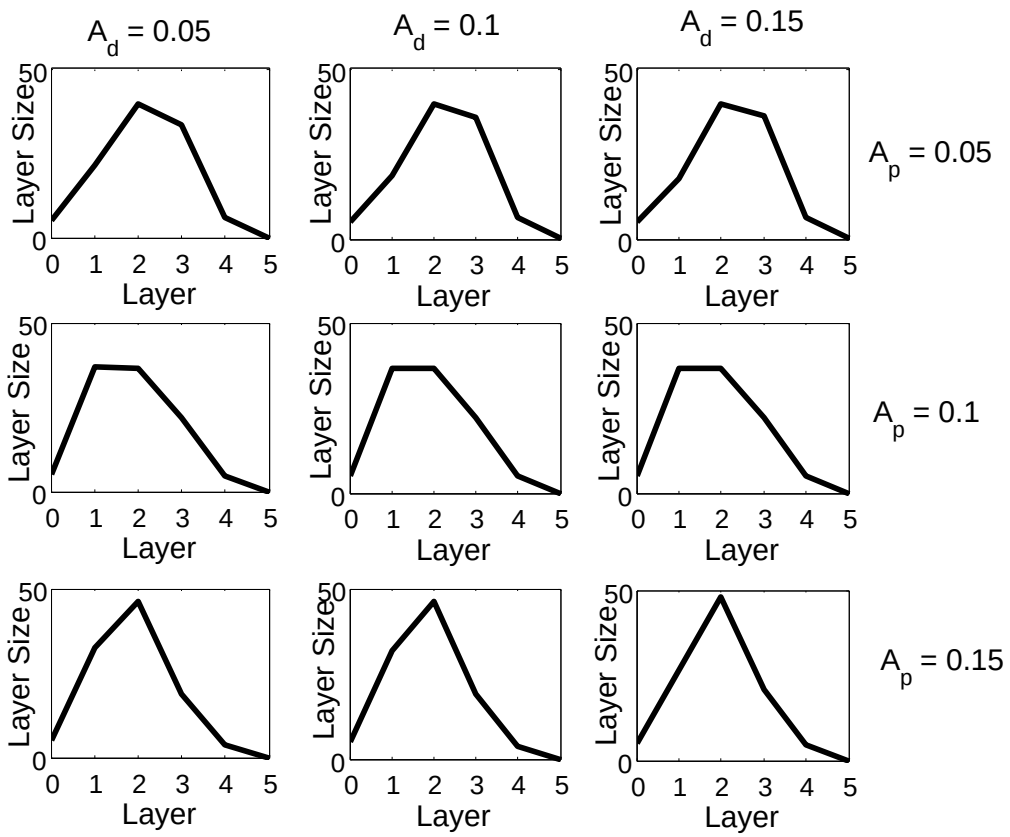


Figure 5.10: The network shape is determined by A_p not A_d. (Results obtained with recurrence relation, Equation (5.18): $\lambda_p = 0.1$Hz, $\lambda_{\text{in}} = 3$Hz, $\theta = 1$, d $= 0.3$, p $= 0.3$, N $=$ 100, $\text{N}_{\text{in}} = 5$.).

predict the final state of simulations, but also, because it necessitated complete description of the system, this made concrete interactions within the network which had previously only been explained intuitively.

When describing the developmental simulations as a random walk, the relationship between the network and input sizes, the STDP parameters, the relative sizes of the layers in the developing chain and the recruitment rates is illuminated. It is now clear that chain development is determined by the relative recruitment rates of each layer, which are determined by the initial conditions of the simulation. Specifically, the most important contributing factor is the number of steps it takes to recruit a neuron. This is jointly determined by the input size, the size of A_p and the relative sizes of τ_p, τ_{d-} and τ_{d+}, which determine p and d.

Whilst this description is useful both as an efficient estimation method and as an aid to understanding the system, it is not clear whether it could increase understanding of real neural systems. Here a number of simplifying assumptions have been made that

reduce the system to a level of complexity where modelling can give real insight. One such simplification is assuming spontaneous spikes follow a Poisson Process. Actually, the literature on such activity more commonly reports network bursts or oscillations [14, 20, 37, 82, 119, 120, 149]. Converting the random walk description here into one where the steps are regular should be fairly simple and would be an interesting extension to this work. If, as one would expect, similar results were obtained with the undeveloped network firing in bursts this would increase the applicability of this model. Additionally, STDP has not been reported in developing systems. It may be the case that STDP is a synaptic adjustment mechanism only. If structural development is dominated by another mechanism then the main result of this chapter, deriving an expectation for the network structure, cannot be applied in the context of real neural systems.

In conclusion, describing synfire chain development as a random walk has given great insight into the simulated development process, specifically, the relationship between simulation parameters and final network structure. However, due to the extremely abstracted nature of this description, further investigation should be undertaken before applying the random walk description as a model of biological development.

Chapter 6

Discussion

Accumulating evidence [8, 70, 98, 124] suggests that synfire chains [2, 3] are an important structure in neural processing. Capable of producing precisely timed neural sequences, synfire chains have been linked to bird song generation [38, 77, 98], control of motor responses in mammals [124, 141, 163] and processing of sensory information [8, 125]. Recently, there has been a considerable amount of research into how synfire chains develop from immature networks [13, 34, 38, 58, 67, 73, 77, 105]. This previous work, summarised in Chapter 1 of this thesis, has converged on a common method for synfire chain development. This method pairs classical STDP with topological constraints such as pruning or heterosynaptic competition. In chapter 2, such a model was developed which combined classical STDP with pruning of strong synapses. Experimentation with this model provided many useful insights most importantly highlighting the importance of topological constraints. Chapter 3 proposed an alternative way to grow synfire chains, triphasic STDP. The novel development method was examined in a network of binary neurons and it was found that triphasic STDP removed the necessity for topological constraints. However, additional mechanisms were also required. For larger networks to develop it was necessary to add slow depression at synapses which experienced uncorrelated spiking. To maintain the stability of existing chains neurons were require to display activity-dependent excitability. In Chapter 4 this novel method for synfire chain development was tested in a network of leaky integrate and fire neurons. Triphasic STDP also successfully lead to synfire chain development in this more realistic neural model. Finally, in Chapter 5, development was described as a random walk. This simple description illuminated the

key relationships between layer size, development rates and the final network structure.

An important aspect of any study involving STDP is how the STDP function is applied. Through experimentation with the initial model which included classical STDP and pruning (Chapter 2) it was found that the method of STDP application (additive or multiplicative, nearest neighbour or all-to-all) had little effect on the final result with chains forming under all conditions. Previous work has suggested that these parameters were important, both to align with experimental results [74, 86, 113] and to create self-stabilising synaptic weights [52, 160]. This result highlights that statistics of neural firing can have great impact on network behaviour and suggest that previous results based on the assumption of Poisson firing statistics must be reevaluated in systems with regular or complex firing patterns.

The firing pattern of both the developed and developing network was critically important in all models presented here. Sparse spontaneous and higher frequency input firing are necessary to develop chains both with classical STDP and triphasic STDP. Chains that developed through triphasic STDP were particularly sensitive to spontaneous activity post development. Understanding the link between activity and development is a topic of much experimental research *in vivo* [49, 89] and *in vitro* [26, 103]. However, this appears to be an area perfectly suited to computational investigations. Here only rate variations of very simple Poisson and regular activity patterns were explored. In biological systems complex patterns [14, 37] including waves [20, 82] and oscillations [119, 120, 149] are frequently observed. A systematic study into the structures which can be developed through STDP rules plus complex activity patterns would help direct future experimental research and possibly uncover hitherto unconsidered neural structures.

Activity dependent excitability was introduced to stabilise the chains developed through triphasic STDP. The necessity for this addition, although biologically motivated [109, 150, 154, 165], is a drawback of the model. Synaptic transmission is a biological process and therefore stochastic. Neural networks need to be robust enough to cope with some stochastic firing. The fragility of these networks may be a question of scale, for computational reasons, the networks in this study were small and individual plasticity steps large compared to biological networks. It is possible that with much smaller steps in much larger networks, single steps would not have such a destabilising effect. However, recent work [86, 113] suggests that this may not be the case. There STDP driven development was observed in very large networks of leaky integrate and fire neurons and "pathological states" (very high firing rates containing repeating patterns) were reported. Although triphasic STDP was not used there, these states resembled those found here when activity dependent plasticity was deactivated, or when a short refractory period was used. In this

model, such behaviour always originated from a loop within the network. Using neurons with activity dependent excitability was a way to prevent such loops. In large balanced networks it may be possible to prevent recurrent activity through carefully planned inhibition. In addition, triphasic STDP as proposed here removes loops and might help prevent the pathological states. Investigating the effect of triphasic STDP in balanced networks is an interesting topic for future research.

Indeed, in this study, the object was to create feed-forward structures. To do this any recurrence must be suppressed by the developmental mechanisms. If loops did persist, they caused collapse of the chain. However, the brain is not entirely feed-forward; recurrent structures are thought to be important for maintaining activity patterns (in working memory for example [17, 18]). Understanding how different neural structures occur and how the brain differentiates between correct and incorrect development at the microcircuit level is an interesting and open question. Perhaps, the myriad of STDP rules hold some clues?

In the majority of the work undertaken here, one single chain was developed from a fully connected network of neurons. Where multiple input groups existed, many chains could be embedded into one network. However, one large chain always dominated. Experimental work suggests that functional networks could be constructed from many smaller chains [70, 125]. Computational work also supports this suggestion [55, 76]. The larger spatial and temporal timescales between chains would suggest that building such hierarchical structures would require plasticity on completely different timescales and thus rely on different plasticity mechanisms. There is much work still to be done on understanding the interactions between mechanisms which can build individual chains, those that can limit chain development and those that can link structures together.

The triphasic STDP function used in this study had a potentiation window of between seven and eight milliseconds. The triphasic STDP rules observed experimentally have potentiation windows of eight [32], fifteen [116] and twenty five milliseconds [167]. As the transmission delay between layers here is simply the transmission delay between individual neurons, the potentiation windows of the later studies with longer durations would cause the chains to collapse in the model, just as seen with classical STDP. However, in a synfire chain of realistic neural construction it is unlikely that the delay between layers is as simple the transmission delay from one neuron to the next. The distribution of synaptic weights [54] and the synaptic dynamics [147] both effect transmission between layers. As does the background activity of the surrounding network [85, 147]. In addition, an inhibitory component [5, 139] to the network can determine the speed of propagation along the chain. Through this simple model, the basic principles of synfire development

through triphasic STDP are set out: potentiation in the timescale of transmission from one layer to the next, depression otherwise. However, the specifics of these timings are likely to vary from system to system and one would expect to see corresponding variations in the observed STDP timescales.

If, as predicted, the time delays between layers in a biological synfire chain are several times slower than the transmission delay (of the order of twenty milliseconds) then triphasic STDP with aligned timescales would be sufficient for chain development and stability. Experimental observations of classical STDP are usually fitted to a exponential function which gives a strictly positive weight change (potentiation) for all positive spike-time differences. However the recorded results [28] often show no potentiation at all for large positive spike-time differences. If classical STDP was modelled as strictly zero for large spike-time differences, it is possible that in a system with long inter-layer delays the problem of progressive collapse which is due to the long exponential tail (described in Chapter 2) would not occur. It would be likely that, without a second depressive window, another depressive mechanism may have to be employed to remove erroneous far forward connections. It may, however, be possible to relax the strong topological constraints that so restrict the previous developmental models.

The leaky integrate and fire neural model used here emits a single spike in response to brief stimulation. The neurons in song bird HVC (an area in which some believe synfire chains could be found [98]) are bursting [111], they produce many spikes following stimulation. One possible step towards biological realism would be a model incorporating a bursting neurons. The behaviour of the STDP rule in such a model would be of particular interest. It may be the case that the calcium transient, on which STDP is thought to be based, changes with much slower time scales than individual spikes within a burst. In which case, it may be more realistic to model synaptic weight changes as a function of burst-time difference not spike-time difference.

The final chapter of this thesis described the successful development of chains as a 1-D random walk. This simple, yet informative description of synfire chain development highlighted the relationship between layer size, recruitment rate and network structure and led to a recurrence relation which neatly predicts the dimensions of the final chain from the simulation parameters. Forming a closed-form mathematical description of the development advances knowledge of the system from the intuitive explanations gained by observing simulations to a concrete description which can be solved quickly for any parameter set.

This thesis describes a novel and hitherto untested method for developing synfire chains from fully connected neural networks. In the continued absence of a complete

neural wiring diagram of HVC and other areas, this work offers a way to strengthen the synfire chain hypothesis. As a result of this thesis, experimental observation of triphasic STDP would be further evidence for the existence of synfire chains. Experimental validation of the model presented here should include both observations of triphasic STDP and consistent inter-layer transmission times. This work on the development of synfire chains has possible implications for computational neuroscientists working in the areas of development, temporal coding and neural structures as well as offering a testable hypothesis for experimental neuroscientists. In addition, it is hoped that others will find inspiration to also examine the other, less intensively studied, STDP rules.

This research has raised many further interesting research questions which could be pursued in the future. For example, what would be the consequence of including inhibition in the networks? Could an inhibitory component in the system allow multiple chains to develop in a more balanced way? If so, could lateral inhibition between chains allow production of multiple activation patterns (motifs) by the same set of chains? Another interesting question leads from the work on heterogeneous delays. Would it be possible to maintain synfire braids if axonal delays were defined separately from dendritic delays? If so, how would the sparsity of the network effect the stability of the braids? The work on the rate of spontaneous firing introduced the possibility of employing patterned spontaneous activity. Using more realistic spike distributions that often include correlations could result in more complex development trajectories. In addition, much more research lies within the other STDP rules; it may be possible to grow other interesting and stable network structures by application of perhaps symmetric or anti-Hebbian rules.

Bibliography

[1] L. F. Abbott. Synaptic plasticity: taming the beast. *Nature Neuroscience*, 3:1178–1183, 2000.

[2] M. Abeles. *Local Cortical Circuits: An Electrophysiological Study (Studies of Brain Function)*. Springer-Verlag, Berlin, New York, 1982.

[3] M. Abeles. *Corticonics: Neural Circuits of the Cerebral Cortex*. Cambridge University Press, Cambridge, 1991.

[4] M. Abeles, H. Bergman, E. Margalit, and E. Vaadia. Spatiotemporal firing patterns in the frontal cortex of behaving monkeys. *Journal of Neurophysiology*, 70:1629–1638, 1993.

[5] Y. Aviel, D. Horn, and M. Abeles. Synfire waves in small balanced networks. *Neurocomputing*, 58-60:123–127, 2004.

[6] Y. Aviel, D. Horn, and M. Abeles. Memory capacity of balanced networks. *Neural Computation*, 17:691, 2005.

[7] Y. Aviel, C. Mehring, M. Abeles, and D. Horn. On embedding synfire chains in a balanced network. *Neural Computation*, 15:1321–1340, 2003.

[8] I. Ayzenshtat, E. Meirovithz, H. Edelman, U. Werner-Reiss, E. Bienenstock, M. Abeles, and H. Slovin. Precise spatiotemporal patterns among visual cortical areas and their relation to visual stimulus processing. *Journal of Neuroscience*, 30:11232–11245, 2010.

[9] S. N. Baker and R. N. Lemon. Precise spatiotemporal repeating patterns in monkey primary and supplementary motor areas occur at chance levels. *Journal of Neurophysiology*, 84:1770–1780, 2000.

[10] C. Bell, V. Han, Y. Sugawara, and K. Grant. Synaptic plasticity in a cerebellum-like structure depends on temporal order. *Nature*, 387:278–281, 1997.

[11] 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*, 18:10464–10472, 1998.

[12] G.-Q. Bi and M.-M. Poo. Synaptic modification by correlated activity: Hebb's postulate revisited. *Annual Review of Neuroscience*, 24:139–166, 2001.

[13] E. Bienenstock. Notes on the growth of a "composition machine". In *Interdisciplinary Workshop on Compositionality in Cognition and Neural Networks*, 1991.

[14] A. Blankenship. Mechanisms underlying spontaneous patterned activity in developing neural circuits. *Nature Reviews Neuroscience*, 11:18, 2010.

[15] M. Bolduc and M. Levine. A review of biologically motivated space-variant data reduction models for robotic vision. *Computer Vision and Image Understanding*, 69:170–184, 1998.

[16] M. Brainard and A. Doupe. Interruption of a basal ganglia-forebrain circuit prevents plasticity of learned vocalizations. *Nature*, 404:762–766, 2000.

[17] N. Brunel. Dynamics and plasticity of stimulus-selective persistent activity in cortical network models. *Cerebral Cortex*, 13:1151–1161, 2003.

[18] 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*, 11:63–85, 2001.

[19] D. Buonomano. A learning rule for the emergence of stable dynamics and timing in recurrent networks. *Journal of Neurophysiology*, 94:2275–2283, 2005.

[20] D. Butts. Retinal waves: Implications for synaptic learning rules during development. *The Neuroscientist*, 8:243, 2002.

[21] M. Butz, F. Wörgötter, and A. van Ooyen. Activity-dependent structural plasticity. *Brain Research Reviews*, 60:287–305, 2009.

[22] E. Callaway. Cell type specificity of local cortical connections. *Journal of Neurocytology*, 31:231–237, 2002.

[23] N. Caporale and Y. Dan. Spike timing-dependent plasticity: A hebbian learning rule. *Annual Review of Neuroscience*, 31:25–46, 2008.

[24] J. Chapin. Using multi-neuron population recordings for neural prosthetics. *Nature Neuroscience*, 7:452–455, 2004.

[25] Z. Chi and D. Margoliash. Temporal precision and temporal drift in brain and behavior of zebra finch song. *Neuron*, 32:899–910, 2001.

[26] E. Cohen, M. Ivenshitz, V. Amor-Baroukh, V. Greenberger, and M. Segal. Determinants of spontaneous activity in networks of cultured hippocampus. *Brain Research*, 1235:21–30, 2008.

[27] Y. Dan and M.-M. Poo. Spike timing-dependent plasticity of neural circuits. *Neuron.*, 44:23–30, 2004.

[28] Y. Dan and M.-M. Poo. Spike timing-dependent plasticity: From synapse to perception. *Physiological reviews.*, 86:1033–1048, 2006.

[29] V. Dasika, J. White, and H. Colburn. Simple models show the general advantages of dendrites in coincidence detection. *Journal of Neurophysiology*, 97:3449–3459, 2007.

[30] D. Debanne, B. Gahwiler, and S. Thompson. Long-term synaptic plasticity between pairs of individual ca3 pyramidal cells in rat hippocampal slice cultures. *The Journal of Physiology*, 507:237–247, 1998.

[31] E. Debski and H. Cline. Activity-dependent mapping in the retinotectal projection. *Current Opinion in Neurobiology*, 12:93–99, 2002.

[32] J. Delgado, J. Gmez-Gonzlez, and N. Desai. Pyramidal neuron conductance state gates spike-timing-dependent plasticity. *The Journal of Neuroscience*, 30:157–15725, 2010.

[33] M. Diesmann, M.-O. Gewaltig, and A. Aertsen. Stable propagation of synchronous spiking in cortical neural networks. *Nature*, 402:529–533, 1999.

[34] R. Doursat and E. Bienenstock. Neocortical self-structuration as a basis for learning. In *5th International Conference on Development and Learning (ICDL 2006)*, 2006.

[35] D. Durstewitz and J. Seamans. Neurocomputational models of working memory. *Nature Neuroscience*, 3:1184–1191, 2000.

[36] R. Faulkner, L. Low, and H.-J. Cheng. Axon pruning in the developing vertebrate hippocampus. *Developmental Neuroscience*, 29:6–13, 2007.

[37] M. Feller. Spontaneous correlated activity in developing neural circuits. *Neuron*, 22:653–656, 1999.

[38] I. Fiete, W. Senn, C. Wang, and R. Hahnloser. Spike-time-dependent plasticity and heterosynaptic competition organize networks to produce long scale-free sequences of neural activity. *Neuron*, 65:563–576, 2010.

[39] E. Fino, V. Paille, J. M. Deniau, and L. Venance. Asymmetric spike-timing dependent plasticity of striatal nitric oxide-synthase interneurons. *Neuroscience*, 160:744–754, 2009.

[40] R. Fonseca, R. Morris, and T. Bonhoeffer. Competing for memory: Hippocampal ltp under regimes of reduced protein synthesis. *Neuron*, 44:1011–1020, 2004.

[41] R. Fonseca, U. Ngerl, and T. Bonhoeffer. Neuronal activity determines the protein synthesis dependence of long-term potentiation. *Nature Neuroscience*, 9:478–480, 2006.

[42] R. Froemke. Spike-timing-dependent synaptic modification induced by natural spike trains. *Nature*, 416:433–438, 2002.

[43] R. Froemke, M.-m. Poo, and Y. Dan. Spike-timing-dependent synaptic plasticity depends on dendritic location. *Nature*, 434:221–225, 2005.

[44] M. Fu and Y. Zuo. Experience-dependent structural plasticity in the cortex. *Trends in Neurosciences*, 34:177–187, 2011.

[45] C. Gardiner. *Handbook of Stochastic Methods*. Springer-Verlag, Berlin Heidelberg, third edition, 2004.

[46] J. Gautrais and S. Thorpe. Rate coding versus temporal order coding: a theoretical approach. *Biosystems*, 48:57–65, 1998.

[47] W. Gerstner and R. Naud. How good are neuron models? *Science*, 326:379–380, 2009.

[48] M.-O. Gewaltig, M. Diesmann, and A. Aertsen. Propagation of cortical synfire activity: survival probability in single trials and stability in the mean. *Neural Networks*, 14:657–673, 2001.

[49] E. D. Gireesh and D. Plenz. Neuronal avalanches organize as nested theta- and beta/gamma-oscillations during development of cortical layer 2/3. *Proceedings of the National Academy of Sciences*, 105:7576–7581, 2008.

[50] M. Graupner and N. Brunel. Mechanisms of induction and maintenance of spike-timing dependent plasticity in biophysical synapse models. *Frontiers in Computational Neuroscience*, 4:136, 2010.

[51] S. Grun. Data-driven significance estimation for precise spike correlation. *Journal of Neurophysiology*, 101:1126–1140, 2009.

[52] R. Gütig, R. Aharonov, S. Rotter, and H. Sompolinsky. Learning input correlations through nonlinear temporally asymmetric hebbian plasticity. *The Journal of Neuroscience*, 23:3697–3714, 2003.

[53] R. Hahnloser, A. Kozhevnikov, and M. Fee. An ultra-sparse code underliesthe generation of neural sequences in a songbird. *Nature*, 419:65–70, 2002.

[54] K. Hamaguchi, M. Okada, and K. Aihara. Variable timescales of repeated spike patterns in synfire chain with mexican-hat connectivity. *Neural Computation*, 19:2468–2491, 2007.

[55] A. Hanuschkin, M. Diesmann, and A. Morrison. A reafferent and feed-forward model of song syntax generation in the bengalese finch. *Journal of Computational Neuroscience*, 2011.

[56] D. O. Hebb. *The Organization of Behavior: A Neuropsychological Theory*. Psychology Press [Wiley], Mahwah, NJ [New York], 2002 [1949].

[57] C. Holmgren and Y. Zilberter. Coincident spiking activity induces long-term changes in inhibition of neocortical pyramidal cells. *The Journal of Neuroscience*, 21:8270–8277, 2001.

[58] R. Hosaka, O. Araki, and T. Ikeguchi. Stdp provides the substrate for igniting synfire chains. *Neural Computation*, 20:415–435, 2008.

[59] D. Hubel and T. Wiesel. Receptive fields of single neurones in the cat's striate cortex. *The Journal of Physiology*, 148:574–591, 1959.

[60] D. Hubel and T. Wiesel. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *The Journal of Physiology*, 160:106–154, 1962.

[61] K. Huber, M. Kayser, and M. Bear. Role for rapid dendritic protein synthesis in hippocampal mglur-dependent long-term depression. *Science*, 288:1254–1256, 2000.

[62] C. Huetz, C. Delnegro, K. Lehongre, P. Tarroux, and J. Edeline. The selectivity of canary hvc neurons for the birds own song: Rate coding, temporal coding, or both? *Journal of Physiology-Paris*, 98:395–406, 2004.

[63] J. Huxter, N. Burgess, and O. . J. Keefe. Independent rate and temporal coding in hippocampal pyramidal cells. *Nature*, 425:828–832, 2003.

[64] J. Iglesias, J. Eriksson, F. Grize, M. Tomassini, and A. Villa. Dynamics of pruning in simulated large-scale spiking neural networks. *Biosystems*, 79:11, 2005.

[65] J. Iglesias, J. Eriksson, B. Pardo, M. Tomassini, and A. Villa. Emergence of oriented cell assemblies associated with spike-timing-dependent plasticity. In *Artificial Neural Networks: Biological Inspirations ICANN 2005*, volume 3696, pages 127–132, 2005.

[66] J. Iglesias and A. Villa. Effect of stimulus-driven pruning on the detection of spatiotemporal patterns of activity in large neural networks. *Biosystems*, 89:287–293, 2007.

[67] J. Iglesias and A. Villa. Emergence of preferred firing sequences in large spiking neural networks during simulated neuronal development. *International Journal of Neural Systems*, 18:267–277, 2008.

[68] J. Iglesias and A. Villa. Recurrent spatiotemporal firing patterns in large spiking neural networks with ontogenetic and epigenetic processes. *Journal of Physiology-Paris*, 104:137–146, 2010.

[69] K. Ikeda. A synfire chain in layered coincidence detectors with random synaptic delays. *Neural Networks*, 16:39, 2003.

[70] Y. Ikegaya, . G. Aaron, R. Cossart, D. Aronov, I. Lampl, D. Ferster, and R. Yuste. Synfire chains and cortical songs: Temporal modules of cortical activity. *Science*, 304:559–564, 2004.

[71] Y. Ikegaya, W. Matsumoto, H.-Y. Chiou, R. Yuste, G. Aaron, and H. Tanimoto. Statistical significance of precisely repeated intracellular synaptic patterns. *PLoS ONE*, 3:e3983, 2008.

[72] K. Ishibashi, K. Hamaguchi, and M. Okada. Theory of interaction of memory patterns in layered associative networks. *Journal of the Physical Society of Japan*, 75:114803–114810, 2006.

[73] E. Izhikevich. Polychronization: Computation with spikes. *Neural Computation*, 18:245–282, 2006.

[74] E. Izhikevich and N. Desai. Relating stdp to bcm. *Neural Computation*, 15:1511–1523, 2003.

[75] E. Izhikevich, J. Gally, and G. Edelman. Spike-timing dynamics of neuronal groups. *Cerebral Cortex*, 14:933–944, 2004.

[76] D. Jin. Generating variable birdsong syllable sequences with branching chain networks in avian premotor nucleus hvc. *Physical Review E*, 80:051902, 2009.

[77] J. Jun and D. Jin. Development of neural circuitry for precise temporal sequences through spontaneous activity, axon remodeling, and synaptic plasticity. *PLoS ONE*, 2:e723, 2007.

[78] D. Kantor and A. Kolodkin. Curbing the excesses of youthmolecular insights into axonal pruning. *Neuron*, 38:849–852, 2003.

[79] A. Karni, G. Meyer, C. Rey-Hipolito, P. Jezzard, M. Adams, R. Turner, and L. Ungerleider. The acquisition of skilled motor performance: Fast and slow experience-driven changes in primary motor cortex. *Proceedings of the National Academy of Sciences*, 95:861–868, 1998.

[80] M. Kato and K. Okanoya. Molecular characterization of the song control nucleus hvc in bengalese finch brain. *Brain Research*, 1360:56–76, 2010.

[81] A. Kepecs, M. van Rossum, S. Song, and J. Tegner. Spike-timing-dependent plasticity: common themes and divergent vistas. *Biological Cybernetics*, 87:446–458, 2002.

[82] D. Kerschensteiner and R. Wong. A precisely timed asynchronous pattern of on and off retinal ganglion cell activity during propagation of retinal waves. *Neuron*, 58:851–858, 2008.

[83] K. Kitano, H. Cteau, and T. Fukai. Self-organization of memory activity through spike-timing-dependent plasticity. *Neuroreport*, 13:795–798, 2002.

[84] K. Kitano, H. Okamoto, and T. Fukai. Time representing cortical activities: two models inspired by prefrontal persistent activity. *Biological Cybernetics*, 88:387–394, 2003.

[85] A. Kumar, S. Rotter, and A. Aertsen. Conditions for propagating synchronous spiking and asynchronous firing rates in a cortical network model. *Journal of Neuroscience*, 28:5268–5280, 2008.

[86] S. Kunkel, M. Diesmann, and A. Morrison. Limits to the development of feed-forward structures in large recurrent neuronal networks. *Frontiers in Computational Neuroscience*, 4, 2011.

[87] H. Larralde, P. Trunfio, S. Havlin, and H. Stanley. Territory covered by n diffusing particles. *Nature*, 355:423–426, 1992.

[88] S. le Vay, T. Wiesel, and D. Hubel. The development of ocular dominance columns in normal and visually deprived monkeys. *The Journal of Comparative Neurology*, 191:1–51, 1980.

[89] B. Lendvai, E. Stern, B. Chen, and K. Svoboda. Experience-dependent plasticity of dendritic spines in the developing rat barrel cortex in vivo. *Nature*, 404:876–881, 2000.

[90] A. Leonardo and M. Fee. Ensemble coding of vocal control in birdsong. *The Journal of Neuroscience*, 25:652–661, 2005.

[91] A. Leonardo and M. Konishi. Decrystallization of adult birdsong by perturbation of auditory feedback. *Nature*, 399:466–470, 1999.

[92] R. Lestienne, H. Tuckwell, M. Chalansonnet, and M. Chaput. Repeating triplets of spikes and oscillations in the mitral cell discharges of freely breathing rats. *European Journal of Neuroscience*, 11:3185–3193, 1999.

[93] J. Letzkus, B. Kampa, and G. Stuart. Learning rules for spike timing-dependent plasticity depend on dendritic synapse location. *Journal of Neuroscience*, 26:10420–10429, 2006.

[94] G. Leuba and L. Garey. Comparison of neuronal and glial numerical density in primary and secondary visual cortex of man. *Experimental Brain Research*, 77:31–38, 1989.

[95] W. Levy and O. Steward. Temporal contiguity requirements for long-term associative potentiation/depression in the hippocampus. *Neuroscience*, 8:791–797, 1983.

[96] J. Liu and D. Buonomano. Embedding multiple trajectories in simulated recurrent neural networks in a self-organizing manner. *Journal of Neuroscience*, 29:13172–13181, 2009.

[97] M. Long and M. Fee. Using temperature to analyse temporal dynamics in the songbird motor pathway. *Nature*, 456:189–194, 2008.

[98] M. Long and D. Jin. Support for a synaptic chain model of neuronal sequence generation. *Nature*, 468:394–399, 2010.

[99] A. Luczak, P. Barthó, S. Marguet, G. Buzsáki, and K. Harris. Sequential structure of neocortical spontaneous activity in vivo. *Proceedings of the National Academy of Sciences USA*, 104:347–352, 2007.

[100] L. Luo and D. O'Leary. Axon retraction and degeneration in development and disease. *Annual Review of Neuroscience*, 28:127–156, 2005.

[101] G. M-O and D. M. Nest (neural simulation tool). *Scholarpedia*, 2:1430, 2007.

[102] J. Magee and D. Johnston. A synaptically controlled, associative signal for hebbian plasticity in hippocampal neurons. *Science*, 275:209–213, 1997.

[103] B.-Q. Mao, F. Hamzei-Sichani, D. Aronov, R. Froemke, and R. Yuste. Dynamics of spontaneous activity in neocortical slices. *Neuron*, 32:883–898, 2001.

[104] H. Markram, J. Lbke, M. Frotscher, and B. Sakmann. Regulation of synaptic efficacy by coincidence of postsynaptic aps and epsps. *Science*, 275:213–215, 1997.

[105] N. Masuda and H. Kori. Formation of feedforward networks and frequency synchrony by spike-timing-dependent plasticity. *Journal of Computational Neuroscience*, 22:327–345, 2007.

[106] N. Mataga, Y. Mizuguchi, and T. Hensch. Experience-dependent pruning of dendritic spines in visual cortex by tissue plasminogen activator. *Neuron*, 44:1031–1041, 2004.

[107] P. Melzer, G. Champney, M. Maguire, and F. Ebner. Rate code and temporal code for frequency of whisker stimulation in rat primary and secondary somatic sensory cortex. *Experimental Brain Research*, 172:370–386, 2006.

[108] A. Mokeichev, M. Okun, O. Barak, Y. Katz, O. Ben-Shahar, and I. Lampl. Stochastic emergence of repeating cortical motifs in spontaneous membrane potential fluctuations in vivo. *Neuron*, 53:413–425, 2007.

[109] W. J. Moody and M. M. Bosma. Ion channel development, spontaneous activity, and activity-dependent development in nerve and muscle cells. *Physiological Reviews.*, 85:883–941, 2005.

[110] R. Mooney. Synaptic basis for developmental plasticity in a birdsong nucleus. *The Journal of Neuroscience*, 12:2464–2477, 1992.

[111] R. Mooney. Neurobiology of song learning. *Current Opinion in Neurobiology*, 19:654–660, 2009.

[112] R. Mooney and J. Prather. The hvc microcircuit: The synaptic basis for interactions between song motor and vocal plasticity pathways. *The Journal of Neuroscience*, 25:1952–1964, 2005.

[113] A. Morrison, A. Aertsen, and M. Diesmann. Spike-timing-dependent plasticity in balanced random networks. *Neural Computation*, 19:1437, 2007.

[114] A. Morrison, M. Diesmann, and W. Gerstner. Phenomenological models of synaptic plasticity based on spike timing. *Biological Cybernetics*, 98:459–478, 2008.

[115] S. Musallam, B. Corneil, B. Greger, H. Scherberger, and R. Andersen. Cognitive control signals for neural prosthetics. *Science*, 305:258–262, 2004.

[116] M. Nishiyama, K. Hong, K. Mikoshiba, M.-m. Poo, and K. Kato. Calcium stores regulate the polarity and input specificity of synaptic modification. *Nature*, 408:584–588, 2000.

[117] K. W. Nordeen and E. J. Nordeen. Anatomical and synaptic substrates for avian song learning. *Journal of Neurobiology*, 33:532–548, 1997.

[118] M. O'Donovan. The origin of spontaneous activity in developing networks of the vertebrate nervous system. *Current Opinion in Neurobiology*, 9:94–104, 1999.

[119] M. O'Donovan, N. Chub, and P. Wenner. Mechanisms of spontaneous activity in developing spinal networks. *Journal of Neurobiology*, 37:131, 1998.

[120] T. Opitz, A. de Lima, and T. Voigt. Spontaneous development of synchronous oscillatory activity during maturation of cortical networks in vitro. *Journal of Neurophysiology*, 88:2196–2206, 2002.

[121] M. W. Oram, M. C. Wiener, R. Lestienne, and B. J. Richmond. Stochastic nature of precisely timed spike patterns in visual system neuronal responses. *Journal of Neurophysiology*, 81:3021–3033, 1999.

[122] S. Panzeri and S. Schultz. A unified approach to the study of temporal, correlational, and rate coding. *Neural Computation*, 13:1311–1349, 2001.

[123] T. Poggio and M. Riesenhuber. Hierarchical models of object recognition in cortex. *Nature Neuroscience*, 2:1019–1025, 1999.

[124] Y. Prut, E. Vaadia, H. Bergman, I. Haalman, H. Slovin, and M. Abeles. Spatiotemporal structure of cortical activity: Properties and behavioral relevance. *Journal of Neurophysiology*, 79:2857–2874, 1998.

[125] F. Pulvermller and Y. Shtyrov. Spatiotemporal signatures of large-scale synfire chains for speech processing as revealed by meg. *Cerebral Cortex*, 19:79–88, 2009.

[126] C. Pytte, M. Gerson, J. Miller, and J. Kirn. Increasing stereotypy in adult zebra finch song correlates with a declining rate of adult neurogenesis. *Developmental Neurobiology*, 67:1699–1720, 2007.

[127] G. Rizzolatti, L. Fogassi, and V. Gallese. Motor and cognitive functions of the ventral premotor cortex. *Current Opinion in Neurobiology*, 12:149–154, 2002.

[128] P. Roberts and C. Bell. Spike timing dependent synaptic plasticity in biological systems. *Biological Cybernetics*, 87:392–403, 2002.

[129] S. Royer and D. Par. Conservation of total synaptic weight through balanced synaptic depression and potentiation. *Nature*, 422:518–522, 2003.

[130] J. Rubin, D. Lee, and H. Sompolinsky. Equilibrium properties of temporally asymmetric hebbian plasticity. *Physical Review Letters*, 86:364–367, 2001.

[131] E. Salinas, A. Hernndez, A. Zainos, and R. Romo. Periodicity and firing rate as candidate neural codes for the frequency of vibrotactile stimuli. *The Journal of Neuroscience*, 20:5503–5515, 2000.

[132] J. Sanes and J. Donoghue. Plasticity and primary motor cortex. *Annual Review of Neuroscience*, 23:393–415, 2000.

[133] S. Schrader, S. Grn, M. Diesmann, and G. L. Gerstein. Detecting synfire chain activity using massively parallel spike train recording. *Journal of neurophysiology*, 100:2165–2176, 2008.

[134] D. Schubert, R. Ktter, and J. Staiger. Mapping functional connectivity in barrel-related columns reveals layer- and cell type-specific microcircuits. *Brain Structure and Function*, 212:107–119, 2007.

[135] A. Schwartz. Cortical neural prosthetics. *Annual Review of Neuroscience*, 27:487–507, 2004.

[136] A. Schwartz, X. Cui, D. Weber, and D. Moran. Brain-controlled interfaces: Movement restoration with neural prosthetics. *Neuron*, 52:205–220, 2006.

[137] M. Segal. Dendritic spines and long-term plasticity. *Nature Review Neuroscience*, 6:277–284, 2005.

[138] T. Serre, L. Wolf, and T. Poggio. Object recognition with features inspired by visual cortex. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 2, pages 994–1000, 2005.

[139] T. Shinozaki, M. Okada, A. Reyes, and H. Cteau. Flexible traffic control of the synfire-mode transmission by inhibitory modulation: Nonlinear noise reduction. *Physical Review E*, 81:011913, 2010.

[140] T. Shmiel, R. Drori, O. Shmiel, Y. Ben-Shaul, Z. Nadasdy, M. Shemesh, M. Teicher, and M. Abeles. Neurons of the cerebral cortex exhibit precise interspike timing in correspondence to behavior. *Proceedings of the National Academy of Sciences of the United States of America*, 102:18655–18657, 2005.

[141] T. Shmiel, R. Drori, O. Shmiel, Y. Ben-Shaul, Z. Nadasdy, M. Shemesh, M. Teicher, and M. Abeles. Temporally precise cortical firing patterns are associated with distinct action segments. *Journal of Neurophysiology*, 96:2645–2652, 2006.

[142] P. Sjöström2006 and M. Häusser. A cooperative switch determines the sign of synaptic plasticity in distal dendrites of neocortical pyramidal neurons. *Neuron.*, 51:227–238, 2006.

[143] S. Song, K. Miller, and L. F. Abbott. Competitive hebbian learning through spike-timing-dependent synaptic plasticity. *Nat Neurosci*, 3:919–926, 2000.

[144] S. Song, P. Sjöström, M. Reigl, S. Nelson, and D. Chklovskii. Highly nonrandom features of synaptic connectivity in local cortical circuits. *PLoS Biology*, 3:e68, 2005.

[145] D. Standage, S. Jalil, and T. Trappenberg. Computational consequences of experimentally derived spike-time and weight dependent plasticity rules. *Biological Cybernetics*, 96:615–623, 2007.

[146] E. Stark and M. Abeles. Unbiased estimation of precise temporal correlations between spike trains. *Journal of Neuroscience Methods*, 179:90–100, 2009.

[147] M. Stimberg, T. Hoch, and K. Obermayer. The effect of background noise on the precision of pulse packet propagation in feed-forward networks. *Neurocomputing*, 70:1824–1828, 2007.

[148] E. T. Stoneham, E. M. Sanders, M. Sanyal, and T. C. Dumas. Rules of engagement: Factors that regulate activity-dependent synaptic plasticity during neural network development. *The Biological Bulletin*, 219:81–99, 2010.

[149] J.-J. Sun and H. Luhmann. Spatio-temporal dynamics of oscillatory network activity in the neonatal mouse cerebral cortex. *European Journal of Neuroscience*, 26:1995–2004, 2007.

[150] M. Syed, S. Lee, J. Zheng, and Z. Zhou. Stage-dependent dynamics and modulation of spontaneous waves in the developing rabbit retina. *Journal of Physiology*, 560:533–549, 2004.

[151] Y. Takahashi, H. Kori, and N. Masuda. Self-organization of feed-forward structure and entrainment in excitatory neural networks with spike-timing-dependent plasticity. *Physical Review E*, 79:051904, 2009.

[152] T. Tetzlaff, A. Morrison, T. Geisel, and M. Diesmann. Consequences of realistic network size on the stability of embedded synfire chains. *Neurocomputing*, 58-60:117, 2004.

[153] A. Tindell, J. Aldridge, and K. Berridge. Ventral pallidal representation of pavlovian cues and reward: Population and rate codes. *Journal of Neuroscience*, 24:1058–1069, 2004.

[154] N. X. Tritsch and D. E. Bergles. Developmental regulation of spontaneous activity in the mammalian cochlea. *Journal of Neuroscience*, 30:1539–1550, 2010.

[155] A. Turner and W. Greenough. Differential rearing effects on rat visual cortex synapses. i. synaptic and neuronal density and synapses per neuron. *Brain Research*, 329:195–203, 1985.

[156] T. Tzounopoulos, Y. Kim, D. Oertel, and L. Trussell. Cell-specific, spike timing-dependent plasticities in the dorsal cochlear nucleus. *Nat Neurosci*, 7:719–725, 2004.

[157] T. Tzounopoulos and N. Kraus. Learning to encode timing: Mechanisms of plasticity in the auditory brainstem. *Neuron*, 62:463–469, 2009.

[158] N. van Kampen. *Stochastic processes in physics and chemistry*. North Holland, Amsterdam, 1992.

[159] M. van Rossum, G. Turrigiano, and S. Nelson. Fast propagation of firing rates through layered networks of noisy neurons. *Journal of Neuroscience*, 22:1956–1966, 2002.

[160] M. C. W. van Rossum, G. Q. Bi, and G. G. Turrigiano. Stable hebbian learning from spike timing-dependent plasticity. *Journal of Neuroscience*, 20:8812–8821, 2000.

[161] R. van Rullen and S. Thorpe. Rate coding versus temporal order coding: what the retinal ganglion cells tell the visual cortex. *Neural computation*, 13:1255–1283, 2001.

[162] A. Villa and M. Abeles. Evidence for spatiotemporal firing patterns within the auditory thalamus of the cat. *Brain Research*, 509:325, 1990.

[163] A. E. P. Villa, I. V. Tetko, B. Hyland, and A. Najem. Spatiotemporal activity patterns of rat cortical neurons predict responses in a conditioned task. *Proceedings of the National Academy of Sciences USA*, 96:1106–1111, 1999.

[164] H.-X. Wang, R. Gerkin, D. Nauen, and G.-Q. Bi. Coactivation and timing-dependent integration of synaptic potentiation and depression. *Nat Neurosci*, 8:187–193, 2005.

[165] D. K. Warland, A. D. Huberman, and L. M. Chalupa. Dynamics of spontaneous activity in the fetal macaque retina during development of retinogeniculate pathways. *Journal of Neuroscience*, 26:5190–5197, 2006.

[166] M. Weliky and L. C. Katz. Correlational structure of spontaneous neuronal activity in the developing lateral geniculate nucleus in vivo. *Science*, 285:599–604, 1999.

[167] G. M. Wittenberg and S. S.-H. Wang. Malleability of spike-timing-dependent plasticity at the ca3-ca1 synapse. *Journal of Neuroscience*, 26:6610–6617, 2006.

[168] M. Woodin, K. Ganguly, and M.-M. Poo. Coincident pre- and postsynaptic activity modifies gabaergic synapses by postsynaptic changes in cl transporter activity. *Neuron*, 39:807–820, 2003.

[169] H. Yamahachi, S. Marik, J. McManus, W. Denk, and C. Gilbert. Rapid axonal sprouting and pruning accompany functional reorganization in primary visual cortex. *Neuron*, 64:719–729, 2009.

[170] A. Yu and D. Margoliash. Temporal hierarchical control of singing in birds. *Science*, 273:1871–1875, 1996.

[171] J.-C. Zhang, P.-M. Lau, and G.-Q. Bi. Gain in sensitivity and loss in temporal contrast of stdp by dopaminergic modulation at hippocampal synapses. *Proceedings of the National Academy of Sciences of the United States of America*, 2009.

[172] L. Zhang, H. Tao, C. Holt, W. Harris, and M.-m. Poo. A critical window for cooperation and competition among developing retinotectal synapses. *Nature*, 395:37–44, 1998.

[173] B. lveczky and T. Gardner. A bird's eye view of neural circuit formation. *Current Opinion in Neurobiology*, 21:124–131, 2011.