INF/SCR-04-15

Spike timing dependent structural plasticity

in a single model neuron

Matthijs van Leeuwen

Intelligent Systems Group,
Institute for Information and Computing Sciences, Utrecht University
Correspondence e-mail address: mleeuwen@cs.uu.nl

Universiteit Utrecht

Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Submitted in partial fulfilment of the requirements
for the degree of Master of Science

Student
Matthijs van Leeuwen

Period
September 2003 - June 2004

Supervisors
Intelligent Systems Group,
Utrecht University, the Netherlands
Prof. Dr. John-Jules Ch. Meyer
Dr. Marco A. Wiering

Institute for Neuroinformatics,
Eidgenössische Technische Hochschule Zürich &
Universität Zürich, Switzerland
Prof. Dr. Rodney Douglas
Dr. Ora Ohana

Abstract

We explored the possibilities of structural spike timing dependent plasticity (STDP) in a new discrete model neuron. The Adaptive Neuron consists of a soma represented by a regular Integrate-and-Fire unit, a dendritic tree represented by a graph and synapses attached to the dendrites. Synapses observe excitatory postsynaptic potentials and back-propagating action potentials, the resulting spike timings are used by STDP rules to control both synaptic weight and location.

It is shown that synapses can be ordered on a linear dendrite according to a sequence in the stochastic input spike trains using Location STDP rules. This sequence detection is not very robust to noise when using only spike timings: stochastic synapses, noisy sequences or adding synapses with independent Poisson inputs badly degrades performance. If we allow STDP to use a distance measure through the amplitude of the back-propagating AP, synapses are ordered in the reverse sequence and performance is less liable to noise. Weight STDP helps Location STDP with sequence detection, as it allows the model to filter synapses that are correlated with the output. Discrimination between two groups of correlated synapses is possible through branching based on spike timings, whereas discrimination between correlated and uncorrelated synapses is possible through another linear dendrite Location STDP using only spike timings.

The results emphasize the importance of structural neural models for exploring neural plasticity and show that STDP rules totally independent of synaptic location on the dendrite have limited abilities.

Acknowledgements

Completing my study, doing the research for this thesis and the actual writing process of this thesis wouldn't have been possible without the help and support of many people.

From November 2003 to March 2004, I spent four months in Zürich (Switzerland), where I had the opportunity to work in the Institute for Neuroinformatics (INI). The many researchers in the institute together explore a very broad range of disciplines in neuroscience and computer science, from neurophysiology via theoretical modelling and analog chip design to artificial neural networks and robotics. It was a great experience for me to work in that environment and see so much of other research during my stay. I would especially like to thank Ora Ohana for her efforts as my supervisor: for introducing me to 'real' neuroscience, giving me the opportunity to do this thesis project, the many discussions, the support with my lab meeting talk and thoroughly proofreading this thesis. It was a real pleasure working with you! Thanks go also out to Rodney Douglas for his advice and the fruitful discussions and to Pratap and Linda for always being very friendly and having me at their place during a later visit.

There were others without whom my stay in Zürich wouldn't have been as much fun as it was. First of all, it was great that Jilles was there in the same period as I was. Weekends were never boring: exploring Zürich and its surroundings, playing table tennis, drinking, cooking and eating together. And of course, discussing our projects resulted in some nice ideas. I often had fun in the student lab as well: I talked, had lunch and went out with Yohei, Andres, Daniel, Christoph and Vincenz. In the student house in the Witellikerstrasse, life was fun because of the presence of Neil, Gülden, Sascha and Lorenz. Thanks to all!

Back to Holland. Two years ago, two good friends and I founded the Adaptive Intelligence Laboratory in Utrecht. The three of us had a great time there: the hundreds of photo's and movies we shot proof this. Also, we together headed out in the direction of biologically inspired artificial intelligence and composed our own study programme. Without Jilles and Arne, I wouldn't have been where I am now. Many thanks go out also to John-Jules Meyer and Marco Wiering, for giving us the opportunity to set up this lab, to allow us to go our own ways and supervising our thesis projects from a distance.

So far about the people directly related to my study, but family and other friends have supported me as well. I would like to especially thank my girlfriend Felien, for supporting me in my decision to go abroad. Thanks go also out to my parents, my sister Karin and Henk & Sabine for the support and electronic 'conversations' we had during my visit to Zürich.

Table of Contents

Chapter 1
Introduction

The brain is fascinating. We can move ourselves in many different ways: walking, swimming and cycling are only a few examples and we can do these things without even thinking about it. We can recognise faces within hundreds of milliseconds, estimate how far objects are, we can read and understand written text. We can recognise odours and tastes. We are aware of ourselves, something we call self-consciousness. Furthermore, we have an incredible memory and one character or visual cue may be enough to recall something that we experienced or learned a long time ago. Apart from all this, we are able to think about anything we can imagine and in the end we can even try to understand how we can do all this.

Our quest for the understanding of the brain is already going on for a long time and many people have made tremendous efforts to get to know as much as possible. But still, despite all efforts, we've hardly started to understand, each small question that we answer results in dozens of new questions. The only thing we can be sure of is that the brain is amazingly complex; more complex than we can possibly understand?

The brain consists of billions of cells, called neurons. These neurons communicate with binary electric signals called spikes or action potentials (APs). In this thesis, we will model a single neuron and it will become clear that the functioning of one such cell is already complicated.

In computer science, artificial neural networks (ANNs) have become a widely accepted and used computational tool since their introduction in the 1940s. An ANN is a model inspired by the nervous system and consists of model neurons. The first generation of neural networks consisted of very simple so-called McCulloch-Pitts threshold neurons [28,32]: in such a neuron, incoming binary signals are weighted, summed and the outgoing binary signal is high if and only if the sum exceeds a certain threshold. By having multiple neurons and connecting them with 'synaptic weights,' a network is created. This basic concept hasn't changed for a long time, although many modifications were made to make neural networks computationally stronger. The second generation of ANNs used continuous activation functions instead of a threshold, which is based on the assumption that real neurons use average firing rates to encode information. Also, recurrent connections allow networks to have some kind of memory [12,23]. In the first two generations, many different types of neural networks have been proposed [32] and numerous learning algorithms, both unsupervised and supervised, are available [54,69]. These networks are commonly used in many applications nowadays, for pattern recognition, signal prediction, signal detection and much more.

It wasn't until the beginning of the 80s that it was recognised that real neurons use more than just average firing rates in their 'computations': real neurons spike and precise firing times may be of great importance. The third generation of neural networks was established with the first spiking neural network (SNN), being biologically more realistic than its predecessors as it uses precise spike times instead of average firing rates. Compared to second genera-

tion networks, SNNs [17,18,67] are relatively new and although much research has been done on this topic, even more work remains to be done. The most common model is the integrate-and-fire (I&F) neuron [33]: incoming signals are integrated and a spike is fired when a certain threshold is reached. Another well known model is Gerstner's Spike Response Model [18].

Although SNNs are accepted by a broad public, research done with SNNs differs greatly in complexity and in realism. In the past decades, models have become more and more realistic, which is also due to advances in neuroscience and applying the newly obtained knowledge in artificial intelligence (AI). But as we will see later, virtually all models represent neurons as single points that are interconnected by synaptic weights, termed point-neurons. However, neurons posses a 3D structure, composed of several compartments. Recent experimental and theoretical research has shown that the existence of these compartments hugely increases the computational power of the neuron, as we will see later.

In neuroscience, highly detailed models exist that simulate the computations performed by realistic multiple-compartment 3D neurons. For example in the NEURON environment [21]. However, they are too detailed and complex to be very useful in artificial intelligence. On the other hand, an increasing number of researchers in AI realise that nature has much to offer and use biology as a source of inspiration. If one pursues a more advanced and biologically more realistic neural network, one should also learn about neuroscience to understand 'the real thing'. The brain is fascinating; why not learn from it?

The main question we will address in this thesis is whether it is possible to achieve structural self-organisation and solve specific computational tasks using spike timing dependent plasticity (STDP). STDP is a recent form of activity dependent synaptic plasticity that uses precise timing between input and output spikes. For this, we will combine current knowledge in neuroscience and artificial intelligence as basis for a new model of the neuron, as we proposed before [30]. Particularly, we will model a single neuron as a treelike structure and thus deviate from the regular point-neuron approach. Two motivations drive this approach: 1. Neurons with widely differing morphologies have evolved in functionally different parts of the nervous system [25] and a likely hypothesis is that each specific morphology serves a specialized function by allowing different types of computation. 2. The tree structure is a useful representation of common information and computer networks, that commence onto a common source. Thus, the model could be used for exploring computations important for neuroscience as well as intelligent systems in computer science.

In the Adaptive Neuron model, a graph representing the dendrites is attached to a leaky integrate-and-fire soma. The dendrites collect inputs from the synapses that may be attached to it and propagates collected signals to the soma. Synapses receive spike trains as input, the output of the soma is also a spike train. STDP rules are used to control both synaptic weight and synaptic location, which hasn't been done before to our best knowledge.

The simulations are divided into two sections: sequence and correlation detection. With the sequence detection task, unsupervised local STDP rules have to order synapses to match the sequence in the input spike trains. Correlation detection is the discrimination between two groups of correlated synapses and between a group of correlated and a group of uncorrelated synapses.

Because this is a computer science thesis, no background knowledge on neuroscience is assumed; the neural basis of plasticity will be given in chapter

2. Chapter 2 deals with real neurons, in chapter 3 we will shift our focus towards neural models that incorporate either synaptic or structural plasticity. In chapter 4, we will describe our aims and goals, after which chapter 5 will describe the Adaptive Neuron model and the experiments we did in full detail. We will conclude with conclusions and discussion in chapters 6 and 7.

Chapter 2

Realistic neurons

One can only develop or understand a model properly if one has appropriate knowledge of what is being modelled. Understanding the functioning of real neurons is therefore crucial for a good understanding of the model we will detail in chapter 5. Those who are already familiar with the basic functioning of a neuron (morphology and signalling) may want to skip 2.1, but sections 2.2 and 2.3 about respectively neural computation and spike timing dependent plasticity contain aspects of the neuron that are slightly more advanced.

2.1 Neural basics

Essentially, a neuron consists of three major components: a soma, dendrites and an axon (see figure 2.1a). The soma could be seen as the 'body' of the cell. Typically, dendrites and axon emanate from the soma as a kind of shoots: they are all neurites and grow from the soma in exactly the same way during development – the longest neurite becomes the axon. The dendrites collect electric signals from other neurons and propagate these to the soma, where all incoming signals are integrated. New signals generated in the soma are propagated through the axon to its terminals, where they are transmitted to other neurons.

A neuron in resting condition maintains a difference in potential between the intra- and extracellular fluids: the resting membrane potential, caused by different ion concentrations, is usually between –60 and –70 mV. This means a

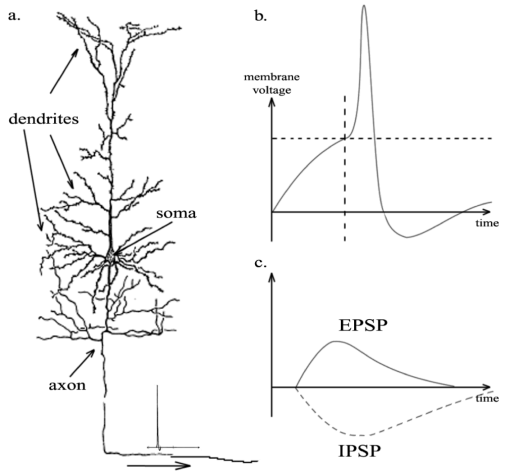

Figure 2.1. (a) The three major compartments of the neuron: the soma, dendrites and the axon. (b) When the membrane potential of the soma reaches a certain threshold, a short all-or-none action potential is generated. After this, a period of hyperpolarisation follows. (c) Typical shapes of an EPSP and an IPSP. [17]

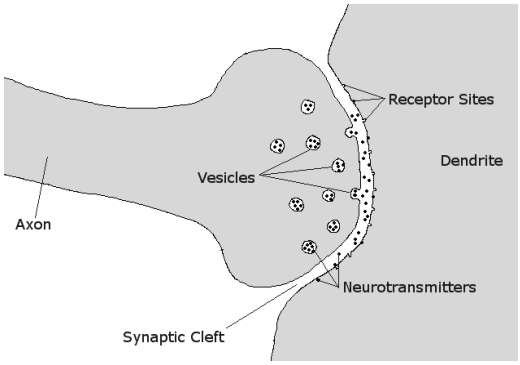

Figure 2.2. The synapse. Presynaptic vesicles of neurotransmitter are released into the synaptic cleft, after which the neurotransmitter binds to postsynaptic receptors that gate ion channels. [web:73]

neuron is polarised in this state. As the cell membrane is the only divider between these fluids, it plays a key role in the electrical functioning of a neuron: altering the membrane properties has a large impact on the membrane potential and the transmission of signals. The membrane is semi-permeable and ions can't flow through it, but ion channels in the membrane make it possible to modify the ion concentrations and therewith the membrane potential. Ion channels are responsible for passive and active signal propagation in dendrites, soma and axon. A local depolarisation of the membrane causes changes in the channel dynamics and results in propagation of the depolarisation. This way, the membrane acts as a RC-circuit with a certain capacitance and resistance, meaning that propagated depolarisations are attenuated along the way.

2.1.1 Neural signalling

So far about resting potential and passive signal propagation, but what causes the initial depolarisation? There are actually two classes of ion channels: one is responsible for passive signal propagation, the other one for signal initiation, active propagation and spiking. The latter class, called gated channels, can be influenced by local conditions: ion concentrations, membrane potential or the presence of chemical messengers. Because of gated voltage channels, a small depolarisation caused by other influences may have large effects: they could amplify a small depolarisation, for example. But other channels can re-polarise the membrane and in the end the membrane potential is always restored to resting potential.

Having this basic knowledge on channel dynamics, let's consider the basics of electrical signalling between neurons. Axon terminals make close connections with dendrites of other neurons to form synapses (see figure 2.2). A synapse could be described as a chemical conveyer of electric signals from one neuron to another, in general from axon to dendrite. Thus we speak about a presynaptic (axon) and a postsynaptic (dendrite) neuron. When an action potential (AP) is initiated in the presynaptic neuron, it arrives at the axon terminal and releases vesicles of neurotransmitters into the synaptic cleft (which is a very small extracellular area between the two parts of the synapse). The released neurotransmitter is a chemical messenger, it diffuses in the synaptic cleft and binds to receptors on the receiving side of the synapse. These receptors gate ion channels and allow a current flow. An excitatory postsynaptic current (EPSC) causes a depolarisation of the membrane potential, the so-called excitatory postsynaptic potential (EPSP, see figure 2.1c). A synapse can also have receptor-gated channels that cause hyperpolarisation (further polarisation) of the postsynaptic membrane. We call such synapses inhibitory, in which case we speak about inhibitory postsynaptic currents (IPSCs) and inhibitory postsynaptic potentials (IPSPs, see figure 2.1c). The size of the PSP depends on the strength of the synapse: how much neurotransmitter is released, how many receptors are available, etcetera. Besides, synapses have a stochastic nature: on arrival of a presynaptic spike, neurotransmitter is released with a certain probability. Synapses that cause an EPSP for only 50-60% of the presynaptic spikes are commonly reported.

An EPSP is passively propagated through the dendritic tree towards the soma. Passive propagation decreases the amplitude of the EPSP such that distal (= far away from the soma, as opposed to proximal) synapses cause a smaller somatic voltage change than proximal synapses. However, voltage-gated channels in the dendrites may modify the amplitude and kinetics of distal EPSPs. This way, EPSPs generated at these synaptic sites may have similar

amplitudes to those generated at proximal sites when they arrive in the soma. When the voltage crosses a higher threshold, another type of channels may also be opened and a very large EPSP can be generated; even dendritic spiking is possible.

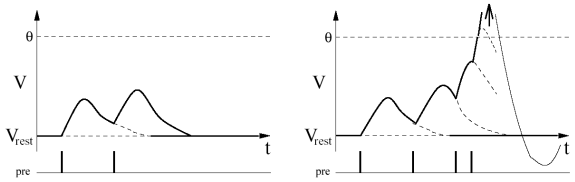

Figure 2.3. Integration of EPSPs in the soma. One or two EPSPs, caused by presynaptic spikes (bottom), are not enough to cause a depolarisation in the soma that exceeds the threshold θ (left). More EPSPs in a short interval do result in an action potential (right). (Adapted from [18].)

The dendrites serve not only as input collectors, but also as sort of a pre-processor: there are many branches in the dendritic tree and EPSPs propagating towards the soma are combined. The morphology and other properties of the dendrites therefore have a large influence on the shape of the voltage response in the soma caused by synaptic inputs. We call this dendritic computation and will be discussed further in the next section.

The soma 'collects' all pre-processed EPSPs and integrates these linearly. It also has a non-linear function though: it has many voltage-gated channels and whenever the membrane potential is depolarized enough to exceed a certain threshold, an all-or-none action potential is generated (see figure 2.1b). An action potential, also called a spike, is a brief but very strong increase of the membrane potential. In many neurons, this threshold can only be reached when many synapses are active at the same time: the voltage increase caused in the soma by a single EPSP is not sufficient (see figure 2.3). An action potential is actually initiated in the region where the axon emanates from the soma, which makes sense as the axon is the 'output device' of the neuron and the action potential is the output of the neuron. It is propagated through the axon to all axonal terminals (which are outgoing synaptic connections to other neurons). Immediately after the AP the soma hyperpolarizes: the voltage is decreased below the resting potential. The soma is in a state of refractoriness, in which it is first impossible (absolute refractoriness) and after that very difficult (relative refractoriness) to initiate a spike. The membrane potential decays back to resting potential and 'regular operation' is resumed. Read further in Kandel et al [25].

2.2 Dendritic computation

We mentioned that the dendritic tree serves as a pre-processor that performs linear computations by combining electric signals in arbours when propagated towards the soma. However, very different dendritic trees exist (see figure 2.4) and it has been observed that they perform non-linear functions as well [28,57,58,66]. These non-linearities are caused by either dendritic morphology or active conductance and may result in complex computation.

If a dendrite would only integrate PSPs linearly, the location of synapses on the dendritic tree wouldn't be very important. The contrary is true though: dendrites shape the voltage response of the soma after a PSP. Let us consider some examples of non-linearities caused by dendritic morphology. First of all, a proximal inhibitory synapse can have a large influence on the propagation of EPSPs: multiple EPSPs that were initiated at more distal synaptic sites can be completely shunted by a single IPSP. A second example is saturation: after

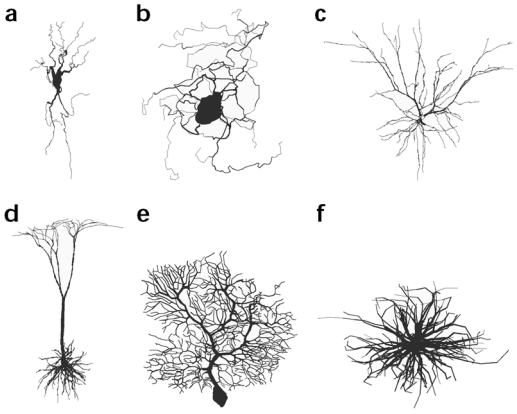

Figure 2.4. Dendritic morphologies. (a) Vagal motoneuron. (b) Olivary neuron. (c) Layer 2/3 pyramidal cell. (d) Layer 5 pyramidal cell. (e) Purkinje cell. (f) α motoneuron. [28]

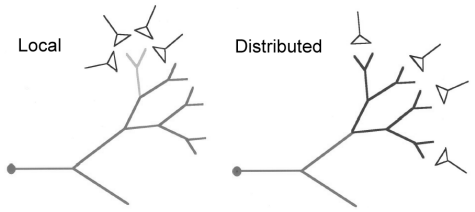

Figure 2.5. Non-linear computations in passive dendrites. The spatial distribution of distal synaptic input sites influences the effect on the potential of the soma. Local distal input easily suffers from saturation and therefore distributed distal input has a larger final effect. [57]

every arbour, the dendrite becomes thinner and is therefore easier saturated (i.e. it cannot propagate more than a certain charge). So when EPSPs are generated by spatially local synapses, this may soon result in saturation, whereas spatially distributed EPSPs are combined further down the tree and may not suffer from this effect (see figure 2.5).

This all changes when the dendrites are active instead of passive: active propagation may amplify EPSPs and dendritic spikes may occur. For example, local EPSPs may result in dendritic spiking, not in saturation. Furthermore, in this case synchronous EPSPs are propagated faster and more efficient than asynchronous EPSPs.

Another important phenomenon is what is often called synaptic scaling. In passive dendrites, EPSPs attenuate as they propagate along the dendrite and thus the amplitude at the origin is larger than the somatic EPSP amplitude. To be able to distinguish these two, we will name these synaptic weight and synaptic efficacy from now on: synaptic weight is the strength of the synapse at its location (initial EPSP amplitude), while synaptic efficacy represents the effect of the generated EPSPs on the soma (somatic EPSP amplitude). With passive propagation, equal synaptic weights result in different synaptic efficacies if the synapses are located on different locations. However, it has been observed that in some neurons, distal synapses are larger in weight, resulting in equal synaptic efficacies independent of location: synaptic scaling [35,59].

Related to this is the overall gain of synaptic input: the number of EPSPs generated in a neuron may vary over time. When hardly any input is received, it may be good to increase the effect of a single EPSP. Decreasing the gain of EPSPs may be useful when there is a lot of input though. Chance et al [8] suggest that the overall level of synaptic input modulates the gain of excitatory inputs, herewith regulating the responsiveness of the neuron.

So far we have mainly discussed aspects that influence EPSP propagation, but dendrites do more: back-propagation of action potentials. APs are axonally initiated and propagated through the axon to other neurons, but it also propagates in the other direction: back through the dendritic tree [28,65,66]. How the shape and size of a back-propagating action potential (BPAP) changes during propagation in the tree depends a great deal on dendritic morphology [66] and on channel densities. In general, an AP is strongly dampened during back-

propagation, but in some types of dendrite the BPAP travels much further than in others. More branching typically results in faster attenuation. BPAPs are of great importance for long-term synaptic plasticity and we'll come back to this later, but also facilitates coincidence detection: a BPAP causes a large depolarisation in the dendrites and this makes it much easier for EPSPs to trigger new dendritic spikes [56], resulting in AP bursts.

After our description of the functioning of real neurons and some examples of dendritic computation, we think it is reasonable to state that there are many factors that influence the 'computations' done by a neuron. Different morphologies and membrane and synapse properties cause great variance between neurons and make neural computation immensely complex. Modelling a neuron as a point-neuron is therefore very unrealistic and we will not take this route. It is now time to address neural plasticity, which makes the neuron even more complex.

2.3 Spike timing dependent plasticity

Before we continue with STDP, it's good to get an impression of what synaptic plasticity is. In this chapter, we restrict ourselves to real neurons and do not discuss any possible applications or models in which STDP has been used; we will do that in the next chapter.

2.3.1 Synaptic plasticity

Synaptic plasticity, in the classical meaning of the words, is the modification of synaptic weight that occurs during the lifetime of a synapse, often influenced by the activity of the pre- and postsynaptic neurons. Synaptic plasticity in real neurons has been studied for a long time and it has been modelled with many rules.

We distinguish two types of plasticity: short and long term. Short term plasticity only has a very short effect, on the time scale of tens of milliseconds and involves time-dependent utilization of the pre-existing resources of the synapse. Long term plasticity, on the other hand, lasts from minutes to days, as it involves modification of the synaptic resources in a lasting manner.

Synaptic plasticity affects the synaptic weight and therewith the synaptic efficacy. Short term plasticity [34,68] is probably mostly caused by facilitation and depletion of neurotransmitter vesicles: a presynaptic AP releases neurotransmitter, but how much is released depends on the number of available vesicles. When many APs arrive soon after each other, the vesicle pool may be depleted and no transmitter can be released, resulting in short term depression (until the vesicle pool is recovered).

Long term synaptic plasticity caused by neural activity results in either long term depression (LTD) or long term potentiation (LTP). Many plasticity rules have been described, but which rules are correct and what the precise mechanisms are is mostly unknown. Long term plasticity that is based on correlations between pre- and postsynaptic firing is often called Hebbian learning, as Hebb was the first to recognise this type of plasticity: he postulated that the synaptic efficacy between two neurons is potentiated when the presynaptic neuron contributes to the firing of the postsynaptic neuron [19]. Correlated pre- and postsynaptic activity should thus result in LTP, whereas uncorrelated activity should result in LTD. Although classical Hebbian learning rules using firing rates have proven to be useful, research in the last ten years has indicated that

precise spike timing is often very important and the term spike timing dependent plasticity has come into existence.

2.3.2 Observed STDP

The hypothesis that correlated pre- and postsynaptic firing induces LTP has proven to be insufficient to explain synaptic plasticity observed in neurons: it sometimes happens that correlated activity results in LTD. This can be explained with spike timing dependent plasticity, which assumes that the relative timing of pre- and postsynaptic spikes determines the changes in synaptic efficacy. Different forms of STDP have been observed, but most common is that presynaptic spikes repetitively preceding postsynaptic spiking invokes LTP, whereas postsynaptic preceding presynaptic spiking invokes LTD [1,5,6,14,16,37,51,62,72]. This is in agreement with both Hebb's postulate and 'common sense': synapses that contribute to the triggering of a neuron should be strengthened, whereas synapses that do not should be weakened. The precise time windows that have been reported differ, but they are all in the size of tens of milliseconds. Also, strongest potentiation occurs when a presynaptic spike precedes a postsynaptic spike by only a few milliseconds and strongest depression occurs when a presynaptic spike immediately follows a postsynaptic spike. A highly asymmetric time window is the result (see figure 2.6).

Generally, the time window extends further along the negative side of the time line than on the positive side. The LTD window is roughly between -5 and –50 ms, the LTP window between +5 and +20 ms (wherein the presynaptic action potential is the reference, t_{post} - t_{pre}). The integral of the LTP side is often smaller than the integral of the LTD side and therefore depression is expected for synapses uncorrelated with the postsynaptic neuron (which seems logical from a stability point of view). However, the peaks of the windows, around –10 and +10 ms, are unequal: the LTP peak is higher, making it 'win' over LTD if all changes are integrated linearly. A difference of only 10 ms in spike timing determines whether a synapse is strengthened or weakened: a very fast transition!

Many other time windows have been observed [4,11], of which a few examples are illustrated in figure 2.7. The windows are very different: the inverse of what we just described has been reported, but also windows with an entirely different shape. Window E depresses all synapses that are correlated with the postsynaptic neuron, while window C only potentiates those and depresses all others. How accurate these windows are and how often they are used by neurons is yet unknown. Window E for example, observed by Egger et al [11], doesn't seem very realistic, as only LTD can only result in full depression of synapses: somehow there must be LTP involved. Apart from this, the functions of the different windows for neurons are largely unknown; simulations with neural models could help here.

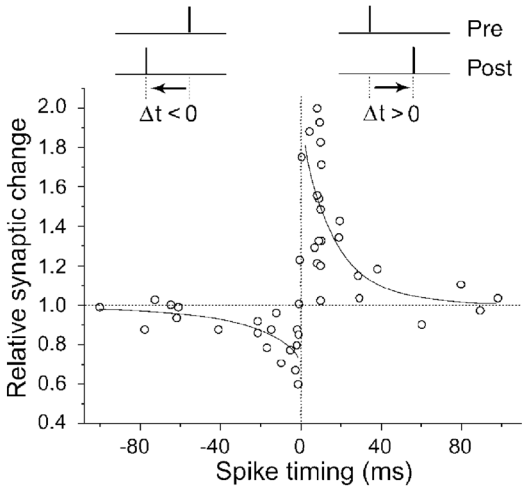

Figure 2.6. Common STDP window. Pre-then-post firing causes LTP, whereas post-then-pre firing causes LTD. The LTP window is narrower, but has a higher peak. [6]

Not only different time windows have been observed for STDP, plasticity also strongly depends on the neuron and synapse type (e.g., STDP doesn't occur in all neuron types) and the amount of LTP is also influenced by the initial synapse efficacy (lower efficacies are potentiated more than higher efficacies) [5]. Another important dependency, which has been left out in many early papers on STDP, is the frequency dependency. The firing rate seems to have a large effect on the modifications made to the synaptic efficacies: with low firing frequencies (below 5 – 10 Hz), LTD is always induced; with high firing frequencies (above 40 – 50 Hz), LTP is always induced; between these boundaries, spike timings and the STDP window determine the weight change [62]. These observations combine the long-known Hebbian learning rules based on firing rates with the more recent spike timing dependent plasticity and resolve the conflicts between these.

So far, we have only considered single spike timings, but natural conditions involve many more: how are weight changes caused by multiple spike timings integrated? There is no definitive answer to this question, as it is still a topic of ongoing debate. And, as with much of what we've seen, it seems that more than one solution may be correct here. The most straightforward way to handle spike timings is to treat them all individually: the effects are independent of each other and integrated linearly. With the common STDP window we started with, the net gain of triplet spiking (pre-post-pre or post-pre-post with equal inter-spike intervals of 10 ms) would be a slight potentiation and this is what we would expect (LTP wins [16,62]). However, other interactions have been proposed to be more realistic: nearest-spike interaction (use only the smallest spike timings) and nearest-spike LTP-wins interaction (if one timing causes LTD and one causes LTP, only the latter is used) [62]. For more natural spike trains, it has been claimed that the first spike timing in a series is dominant [16].

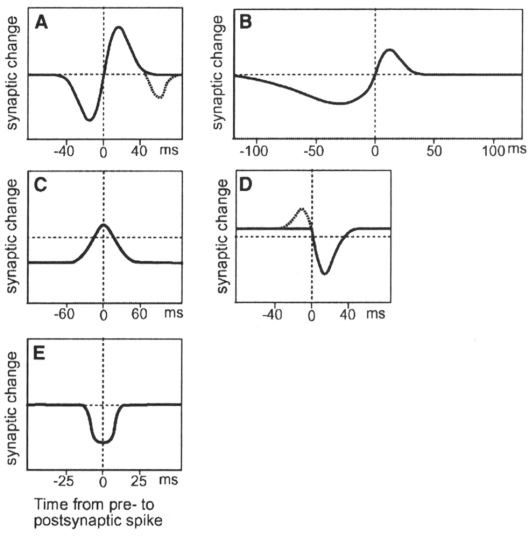

Figure 2.7. Spike timing dependent learning rules, where positive timing indicates that the postsynaptic spike follows the presynaptic spike. (a) Antisymmetric Hebbian learning rule consistent with Markram et al. [37], Zhang et al. [72] and Bi and Poo [5]. A second and later LTD component (dashed line) has been reported in Nishiyama et al. [41]. (b) Antisymmetric learning rule consistent with Feldman [14]. (c) Symmetric Hebbian learning rule. [9] (d) Anti-Hebbian learning rule that is consistent with data presented in Bell et al. [4]. The associative LTP component (dashed) is not statistically significant in vitro, but has been observed in vivo [3]. (e) Symmetric anti-Hebbian learning rule [11]. (Adapted from [51].)

The mechanisms underlying STDP are not yet understood, but it is widely believed that a mechanism exists in the synapse that is capable of detecting the coincidence of EPSPs and BPAP. When an EPSP is initiated somewhere in a synapse in the dendritic tree, it is propagated to the soma, where an AP may be generated. This AP is back-propagated through the dendritic tree and may also arrive at the original synaptic location. The BPAP may then be amplified by the decaying phase of the EPSP and this provides a way of coincidence detection of EPSPs and APs. How this results in LTD/LTP precisely we don't know yet, but it is very probable that a certain type of receptor (NMDAR)

and the intracellular Ca^{2+} concentration play important roles in this in most cases. In particular, the intracellular Ca^{2+} concentration may be an associative signal: an EPSP causes only a small Ca^{2+} influx (which may result in LTD), whereas an EPSP combined with BPAP causes a large Ca^{2+} influx (which may be an intermediate messenger for LTP induction) [36,37,39,65,71].

It's important to note here that STDP is assumed to act locally in a synapse, while all experiments that have been done to show STDP measure membrane potentials in the pre- and postsynaptic somata, not in the synapse itself. Therefore, the timing windows we have seen in this section are the result of spike timings that were recorded in the somata. This seems wrong, as the time that either an EPSP/BPAP needs to be propagated to/from the soma from/to the synapse may be a few milliseconds and we have seen that the transition time between LTD and LTP is only 10 ms. The question that rises is how accurate these STDP windows are and whether there is any dependence on synaptic location.

Another open topic is that of activity-dependent structural changes. Not only synaptic efficacies change during the lifetime of a neuron, dendrites do as well. Most research on structural plasticity is done during the developmental phase of the nervous systems, where growth and chemical influences on growth can be investigated. During this phase, dendrites and synapses sprout, move and retract. But even after this is finished, some dendrites remain plastic and can be influenced by neural activity [70].

Activity-dependent plasticity in neural structure has been reported in literature, especially in spines [13]. Calcium signalling seems to have a large role in neurite morphogenesis [48] and while calcium concentrations are affected by neural activity, activity may indirectly influence morphogenesis. Not only the propagation of EPSPs, but also the back-propagation of action potentials is influenced by morphology [66]. Through structural plasticity, neural activity may be able to shape its own processing and it may be worth investigating whether a structural form of STDP is feasible.

Chapter 3
Learning models

Now that we've seen real world neurons, it is time to take a closer look at existing neural models. Because many different kinds of models exist, we will focus on those models that are relevant for this thesis: in the first section, applications of STDP will be described, the second section deals with structural plasticity models.

3.1 STDP applied

In the introduction, artificial neural networks were described that consist of point-neurons interconnected by synapses. In the first two generations of artificial neural networks, these synapses were modelled by their weights only. Many forms of Hebbian learning have been proposed for these ANNs, especially for those of the second generation [2]. They are commonly applied as local unsupervised learning rules to modify the weights of a network and are useful to achieve activity stabilisation and for tasks like pattern recognition, clustering and auto-association. However, spike timing dependent plasticity cannot be applied to these neural networks, simply because they use average firing rates as signals, not individual spikes.

With the introduction of spiking neural networks, it became possible to apply STDP: spike timings can be simulated. In existing spiking neural networks, such as the Spike Response Model [18], it is fairly easy to implement STDP as a local Hebbian-like learning rule: one only has to choose a learning window and incorporate the rule in the existing model to control the synaptic weights.

STDP has been successfully applied in numerous models. An early example is intrinsic firing rate stabilisation and weight structure formation in SNNs [26,27]. With additive STDP and the regular learning window, the average weight of synapses onto a single spiking neuron could be made to converge always to the same value, herewith normalising the output firing rate. Also, introducing a correlation between some synapses resulted in discrimination between correlated and uncorrelated synapses in synaptic weight: correlated synapses became potentiated, uncorrelated synapses were depressed. However, no strict Hebbian STDP learning was used here: apart from spike timings, non-Hebbian terms were also used, modifying the weights on arrival of each EPSP and/or BPAP with a fixed amount.

Several researchers have shown that STDP with the regular window makes it possible for postsynaptic neurons to spike as early as possible. Song et al [63] describes this as competitive Hebbian learning: if synapses fire shortly after each other, the first synapse to fire is strengthened and wins over the synapses that fire later and those are weakened. The synapses compete for control over postsynaptic spike timing and this results in a bimodal distribution of synaptic

weights: the winning synapses become saturated, the synapses on the losing side become fully depressed. Simulations were done with a single integrate-and-fire neuron with 1000 excitatory and 200 inhibitory synapses, of which only the former were plastic. Of the excitatory synapses, those that were correlated won over the uncorrelated, which is in accord with what we have seen before. Roberts [50] compared similar results he obtained with what he called differential learning with classical conditioning: a response that is at first only evoked after a sequence of serial delayed inputs, may after learning with STDP be triggered shortly after the onset of the sequence. The latency between onset of the stimulus and response is minimised by letting the derivative of the postsynaptic spike activity determine the weight change.

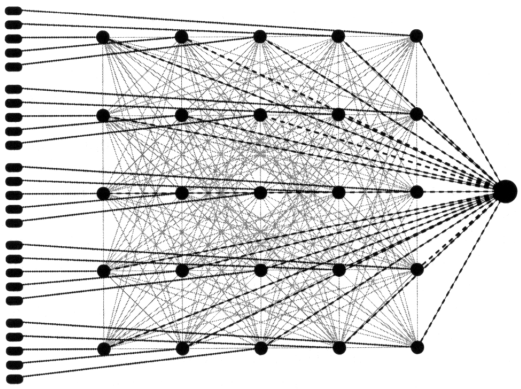

Figure 3.1. Morphology of the model of Nowotny et al [42]. The ovals are artificial input neurons producing rectangular spikes of 3 ms duration at specified times. Each is connected by a non-plastic excitatory synapse to one of the main neurons (dotted lines). The full circles depict the integrate-and-fire neurons. They are connected all-to-all by STDP synapses shown as solid gray lines. The big full circle on the right depicts a neuron with slow calcium dynamics which inhibits all neurons through the non-plastic synapses shown as dashed lines.

A sequence learning application involving the regular learning window has been done with a very fixed network topology [42], see figure 3.1. Each input is connected with a non-plastic synapse to a neuron in the network, basically causing a spike in that neuron each time the input is active. Only one neuron was allowed to be active at a time, the sequences were deterministic and the large inhibitory neuron terminated all network activity after an (in practice) fixed amount of time. All excitatory neurons are interconnected with plastic synapses and as they are always triggered right after each other in a sequence, it is no surprise that the weights of the synapses covering the path of this sequence are strengthened and therefore make it possible for the network to reconstruct these sequences.

Hopfield and Brody [22] state that neurons need a mechanism for self-repair as they continuously deteriorate as a result of noisy activity causing plasticity and the loss and creation of synapses. Using a functioning spiking neural network as starting point, they derive a spike timing dependent learning rule that is able to perform this self-repair. The resulting derived rule is very similar to the regular STDP rule we know, which may be an indication for its function in reality. Moreover, they also show that *de novo* learning is also possible using the same rule.

However, all these experiments considered only point-neurons, disregarding any influence that may come from neural structure. Finding clusters in input vectors and sequence learning profiting from different synaptic delays has been done by Natschläger and Ruf [38]. They used a spiking neural network consisting of integrate-and-fire neurons with many synapses between each two neurons, each with a different delay; the neurons had no real structure, but the

delay effect of dendrites and axons has been captured. A STDP learning window with LTP for small positive timings and LTD for all other timings enabled the network to compute Radial Basis Functions effectively. With inhibition to establish a winner-takes-all mechanism, input vectors with temporal patterns could be clustered. Addition, modification or deletion of clusters during the learning process even resulted in reconfiguration of the network. STDP potentiated the delay lines belonging to a certain cluster and depressed the others. Potentiating multiple delay lines from a single input to an output neuron allowed for variance in the cluster. Although this result is interesting, there are a few drawbacks: the input vectors were rather deterministic and the delay of the back-propagating action potential was the same for all synapses, both not very realistic. Apart from this, the network starts with synapses for all possible delays and filters out those that are useful, which is not very efficient. The experiments Senn [61] did were more realistic, as he proposed that the learning rule should change qualitatively with the firing rate and that unreliable synapses should be used. He also started with many delays and selected from these using STDP.

Rumsey and Abbott [55] investigated a realistic model of a single neuron and tried to achieve synaptic scaling, making synaptic efficacies location-independent using a form of STDP. They succeeded in doing this, but not using strictly Hebbian learning: they increase synaptic weight for each presynaptic spike and decrease weight for negative spike timings. Although this results in effective synaptic scaling, using this non-associative form of LTP and STDP makes it less attractive, as combining this with correlation strengthening or any other application we have seen so far is probably impossible.

Rao and Sejnowski [49] argued that the biophysical implementation of STDP could be Temporal Difference (TD) learning. Using a realistic compartment model of a neuron, they showed that application of TD learning indeed resulted in weight modifications very similar to STDP observed in real neurons. And because TD learning depends on the timings and shapes of BPAPs, the resulting STDP rule is location dependent. Synapses that are further away from the soma observe BPAPs later and in a broader shape, which makes their learning windows broader as well.

We have seen several applications of STDP in neural models: stimulus prediction, discrimination between correlated and uncorrelated inputs, sequence learning, clustering, self-repair and synaptic scaling. However, in most applications neurons are considered to have no structure, despite the fact that structure has a huge influence on plasticity and should not be neglected if one intends to develop a realistic model. Dendritic and axonal delays have been used a few times, but plastic delays haven't been proposed yet.

3.2 Structural plasticity

The impact of active dendrites and structural plasticity on the memory capacity of neural tissue has been investigated theoretically and in simulations by Poirazi and Mel [47]. They used a simple neuron model consisting of a soma, a number of identical dendrite branches and input lines that can be connected to several locations on the dendrites (see figure 3.2). Unsurprisingly, they conclude that non-linear propagation of dendritic inputs results in a larger memory capacity than linear propagation. They used a stochastic gradient descent learning rule to train the neuron to detect certain patterns, modifying both syn-

aptic weights and locations. However, although synapse fixation/removal was activity-dependent, re-location was done in a 'trial-and-error' manner which is not very efficient and learning is supervised.

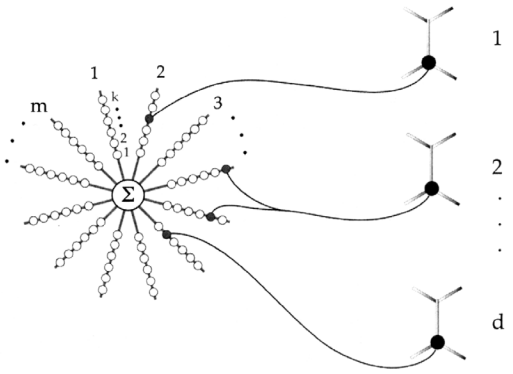

Figure 3.2. Simplified Abstraction of a Dendritic Tree by Poirazi & Mel [47]. Cell is modelled as a set of *m* identical branches connected to a soma, where each branch contains *k* excitatory synaptic contacts. Each synapse is driven by one of *d* input lines and is given a small integer-valued weight.

Although not really structural plasticity, Nielsen [40] did use a realistic approach profiting from neural structure that could be extended to become plastic. His idea is based on so-called 'hot-spots': it seems realistic that real neurons learn only in regions that are 'activated'. He models this by introducing subsets of synapses that are active during sequence learning: the active subset learns the current sequence, while the other subsets are not affected by this. This may be a solution to the stability-plasticity dilemma, making it easier to learn multiple patterns that do not disturb one another. He does this with a second generation recurrent neural network, fixed subsets of synapses and supervised gradient descent learning, but the basic idea could be made plastic and implemented in more realistic models. An algorithm that would optimise the structure to automatically create synaptic subsets would be ideal.

In neuroscience, structural development of neurons has been investigated and simulated as well, but the main focus has mostly been on activity independent growth: for example, how different dendrite morphologies may be developed using chemo taxis. This has also been simulated: there are growing ANNs based on chemo taxis or static patterns [15,60], but these models do not use spike timings. Ooyen et al [43] were among the first to investigate activity dependent neurite outgrowth, which they did in a two dimensional environment with neurons as growing circles. More recently, Van Pelt and Uylings looked on growth from a very computational perspective [45,46]. Hely et al [20] suggested that elongation and branching of neurites uses one and the same mechanism and involves the intracellular concentration of Ca^{2+}. Ramakers et al [48] confirmed this with experiments with real neurons. And as we know by now, $[Ca^{2+}]_i$ is influenced by neural activity and probably involved in STDP. In other words, it's likely that activity influences neural structure.

There is no doubt that dendritic morphology influences activity and plasticity. We have seen this in chapter 2, but simulations confirm this: e.g., the effect of morphology on STDP [64] and firing patterns [44] have been investigated. However, as far as we know, there is no neural model yet that actively uses activity dependent structural plasticity.

Chapter 4

Aims and goals

Our aim is to develop a model of a single neuron that is a compromise between realistic but complicated compartmental neural models and abstract point-neurons lacking any structure. The model neuron should be a tree structure representing the dendrites, to which synapses can be attached. A two or three dimensional tree structure is a large improvement over point-neurons, as it facilitates a natural way to account for propagation delay and signal attenuation and amplification. By making all aspects of the model discrete (signals, time steps), implementation should be relatively simple and simulations computationally not very demanding, unlike compartmental models that include ion channel dynamics. Like ANNs of the third generation, the model's computations will be based on individual spikes, not on average firing rates. Our aim is to develop a model that is computationally simple, yet realistic enough to couple experimental results back to real neurons in a later stage.

We intend to use spike timing dependent plasticity rules inspired by experimentally-observed STDP as local unsupervised learning rules. The rules should operate in the synapses, where they can observe the evoked EPSPs and action potentials back-propagating through the dendritic tree. We will restrict ourselves to purely associative STDP rules.

The challenge at hand is to solve a task relying only on spike timing information. Tasks we will discuss are sequence and correlation detection, which both have been done with STDP before. A few important differences are that we use a discrete structural model as described above and that we modify not only the weights using STDP, but also synaptic locations. Instead of starting with many synapses covering all possible delays, we start with only a few and adapt their delays by moving them on the dendrite in a controlled manner using STDP.

We will explore the possibilities of different STDP rules for controlling both weight and location and the effect of synaptic location on STDP. As we believe neurons are not perfect and often act in a stochastic manner, we aim to reflect this in our simulations to make them more realistic. Both the input spike trains and synapses will be stochastic. An important goal will be to investigate how robust spike timing dependent structural organisation is with respect to noise.

Chapter 5
Results

In the previous chapters, both the biological and theoretical backgrounds of the research have been detailed. We will now proceed with the results of the project. As the neuron model is completely new, we also consider this to be part of the results and explain it in section 5.1. After that, we will give the results of the simulations with uncorrelated Poisson inputs, sequence detection and correlation detection.

5.1 Adaptive Neuron model

The Adaptive Neuron is a discrete model of a single neuron, inspired by both real neurons and existing neural models. Contrary to most models in neuroscience (like the Hodgkin-Huxley model [25]), it doesn't account for realistic electrical signalling, making it computationally much simpler.

Essentially, a neuron is represented by a graph, consisting of nodes and edges (see figure 5.1). The root of the graph is the soma, all other edges and nodes together form the dendrites, emanating from the soma. With this representation, we can model all types of dendritic morphologies with different complexities and at different levels of detail. For example, we could assign three dimensional coordinates to all nodes, but the tree structure itself can already define many different configurations. For simplification, we decided to stick with the latter option: only the structure of the graph determined neural computation, two dimensional coordinates were only calculated for visualisation purposes.

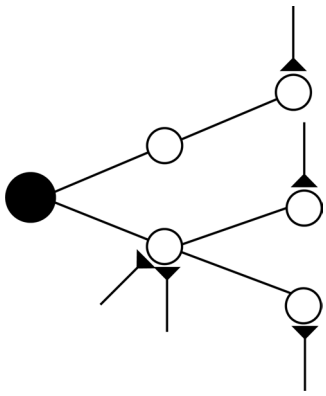

Figure 5.1. Example model neuron. The soma is drawn as a filled black circle, the rest of the graph represents the dendritic tree. Synapses are drawn as black lines with triangles and always attached to one of the dendritic nodes.

In addition to the graph representing the soma and dendrites of the neuron, we modelled synapses as separate entities. A synapse is always attached to a certain dendritic node (not to the soma), but applying a Location STDP rule may result in relocation to other nodes (see 5.1.3). Synapses receive spike trains as input and may generate EPSPs in the dendrite node it is attached to (all synapses we consider are located on the dendrites and the neuron is therefore always postsynaptic, on the receiving side of the synapse). In this thesis, we only consider excitatory synapses: adding inhibitory synapses would make it far more difficult to understand the behaviour of the model.

5.1.1 Neural computation

The model operates in discrete time: running time is divided into time steps and a single update causes the network to advance by 1 ms. In one update, all synapses receive a new input value (spike or no spike) and after computation is finished, the output of the soma (spike/no spike) is updated. Internal computation is now described.

At the start of a network update at time *t*, each synapse receives a new value from its input spike train, in_t (0 or 1). As synapses are stochastic, they may or may not trigger an EPSP in the dendritic node attached to, dependent on release probability $p_{release}$ (default 1). If an EPSP is invoked, the size depends on both the global EPSP scale *e* and individual synaptic weight *w*. The global scale for EPSP size is set manually and depends on the number of synapses and/or firing rates of the input spike trains, as it has a large influence on the output rate (firing rate of the soma). Values between 10 and 40 are common scales. The weight of a synapse is typically a value between 0 and 1 and often changed during a run, as it may be controlled by a Weight STDP rule (see 5.1.4). The output of a synapse *s* at time *t* is now:

$$q_t^s = coin(p_{release}) * w_t * e * in_t \qquad \text{(Eq. 5.1)}$$

In this equation, *coin*(*p*) is a uniform random function that returns 1 with probability *p* and 0 with probability 1 – *p*. Note that the duration of a single EPSP is only one millisecond, a single network update. This makes it possible to treat an EPSP as a single number. EPSPs generated by the synapses are summed and propagated in the dendritic tree, always in the direction of the soma. Forward signal propagation in the dendrites depends only on decay λ (0 < λ < 1, default 0.9) and the charges that are received from the synapses. At time *t*, the charge *Q* of node *n* is:

$$Q_t^n = \lambda\left(\sum_{x \in Nodes(n)} Q_{t-1}^x + \sum_{s \in Syn(n)} q_t^s\right) \qquad \text{(Eq. 5.2)}$$

Where *Nodes(n)* contains all dendritic nodes that are connected to node *n* (but only those further away from the soma) and *Syn(n)* consists of all synapses currently attached to node *n*. The equation shows that EPSPs generated by synapses are immediately added to the charge of the node, but there is a delay of one network update (1 ms) between each two subsequent nodes: at time *t*, a node sums the charges of all child nodes at time *t – 1*. An EPSP that is generated 5 nodes away from the soma, will therefore take 5 ms to arrive in the soma. The decay is applied in every node it travels through, in this example resulting in a decay of λ^5.

This accounts for EPSP propagation in the dendrites, but the soma handles signals differently. The soma is modelled as Leaky Integrate-and-Fire 'neuron': it integrates all incoming charge, has a membrane time constant τ that determines how fast charge is leaked and fires an action potential when the threshold θ is crossed. The membrane potential of the soma *V* and the output of the soma *O* at time *t* are computed as follows:

$$V_t = \begin{cases} V_{rest} + (V_{t-1} + \sum_{x \in Nodes(soma)} Q_{t-1}^x - V_{rest}) e^{\frac{-1}{\tau}} & \text{if } O_{t-1} = 0 \\ V_{rest} + (\sum_{x \in Nodes(soma)} Q_{t-1}^x) e^{\frac{-1}{\tau}} & \text{if } O_{t-1} = 1 \end{cases} \quad \text{(Eq. 5.3)}$$

$$O_t = \begin{cases} 1 & \text{if } V_t > \theta \,\&\, O_{t-1} = 0 \\ 0 & otherwise \end{cases} \quad \text{(Eq. 5.4)}$$

Calculation of the membrane potential in the soma may look quite complicated, but it isn't more than a regular leaky integrate-and-fire computation. Normally, charge coming from the dendrites is added to the potential of the previous time step, after which a leak is applied (an exponential leak, dependent on the difference between V and V_{rest}). When a spike occurred in the previous time step (O_t = 1), the previous potential is disregarded: V is reset to resting potential and new incoming charge is added. The output of the soma O_t is 1 when the membrane potential is higher than the pre-defined threshold and the output of the previous time step was 0 (in other words: we introduced an absolute refractory period of 1 ms), 0 in all other cases. The values we use for the somatic parameters are: V_{rest} = -70 mV, θ = -30 mV and τ = 20 ms.

The firing of an action potential in the soma also initiates the back propagation of this AP. Like the EPSPs, this so-called BPAP travels through the dendritic tree 1 node per time step, but now in the opposite direction. A BPAP starts with strength 1 and decays during back propagation with decay constant κ (default 0.9). And just like an EPSP, it remains in a single node only for the duration of a single time step. The BPAP state of a node n at time t is defined as follows:

$$BPAP_t^n = \kappa * BPAP_{t-1}^{ParentNode(n)} \quad \text{(Eq. 5.5)}$$
$$BPAP_t^{soma} = O_t$$

ParentNode(n) defines the parent node of node n; the node that is connected through an edge to node n in the direction of the soma. The BPAP state of the soma is equal to the output of the soma (and thus always one or zero). Although the BPAP may seem to have no influence on the computation at first sight, this is definitely not the case as its propagation has a large effect on the timing dependent plasticity rules operating locally in the synapses.

So far, we've discussed the basics of our model, but this only considers a static model without plasticity. We will therefore now first explain the STDP rules that we used, after which we'll give an example.

5.1.2 Spike timing dependent plasticity

Most readers have probably already noticed that we distinguish two types of spike timing dependent plasticity rules: 1) rules that modify synaptic weight and 2) rules that relocate synapses. These two types will be described in the next two subsections, but we will now describe some general issues. As we've seen in the previous chapters, many different STDP rules exist: as long as precise timing of pre- and postsynaptic spikes is important for adaptation, we're

dealing with STDP. In a single simulation, we can apply one Weight STDP rule and one Location STDP rule to our synapses. It is actually possible to have different STDP rules for different synapses, but we always used the same rules for all our synapses. Each STDP rule operates independently in each synapse though (i.e., only information that is locally available in the synapse can and may be used by the rule).

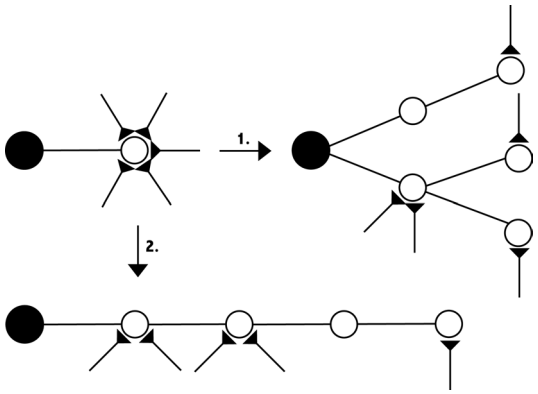

Figure 5.2. Adaptivity examples. The initial configuration is always as shown top left: all synapses on the only dendrite node. STDP relocates the synapses and the dendrite may branch and elongate accordingly. The neuron converged to a branched tree in case 1, while case 2 is an example of a linear dendrite.

With many learning mechanisms, a simulation run is divided into a learning and a test phase. In the learning period, the learning rules are applied and in the test phase, the resulting configuration is tested for a certain task, but without further learning. However, STDP is normally applied as a dynamic and on-the-fly adaptation mechanism that converges to a certain state, but may re-adapt the configuration when the input/output patterns change. This is also how we used STDP: it is applied during the whole run, there is no separate test period.

Each time step, after all signal propagation in the network has been done (both EPSPs and BPAPs), all synapses get the opportunity to evaluate their STDP rules and apply possible changes to weight and/or location. All spike timings $t_{bpap} - t_{epsp}$ that occur are immediately evaluated by both rules. Note that we applied no mechanism like nearest interaction [62], as we found this to introduce more problems than it solved. After evaluation of all timings within a single time step, the rules are allowed to apply their changes: the new synaptic weight can be applied and the synapse can be relocated.

In chapter 2, we mentioned that recent literature suggests that 'traditional' Hebbian learning (rate based LTD and LTP) and the newer spike timing dependent plasticity should be combined into one learning rule. LTD is always the result with firing rates below 5 Hz, LTP always wins with firing rates above 50 Hz and STDP is said to function in between. It is not very obvious how to implement this in a model like the current one, we therefore prefer not to do this and we will look primarily at rates between these boundaries. STDP is always applied, independent of the firing rate.

5.1.3 Location STDP

Location STDP (LSTDP) rules use spike timings and possibly other local information to determine how to relocate synapses. This way, dendritic morphology specific to a dedicated function may evolve: STDP rules are permanently applied during simulation and therefore synapses may constantly be rearranged according to the current input/output patterns. All simulations start with the same basic initial configuration (see figure 5.2), in which all synapses are on the only dendritic node of the neuron. No movement seems possible in the initial configuration, but whenever a synapse is on the outer end of a dendrite and

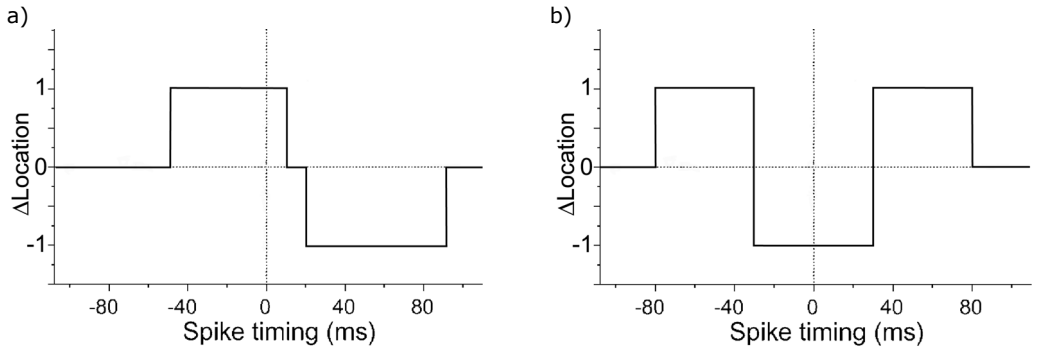

Figure 5.3. a) Box shaped LSTDP. Spike timings between –50 and +10 ms result in relocation away from the soma; spike timings between +20 and +90 ms in relocation towards the soma. b) Hat shaped LSTDP. Small spike timings result in movement towards the soma, larger spike timings in movement away from the soma.

wants to move away from the soma even further, a new node is created and the synapse may move there. A general restriction that applies to all LSTDP rules: a synapse may be relocated only to a neighbouring node in one network update (and may thus not travel multiple nodes at once). In this thesis we will focus primarily on instances of our model that have only a single linear dendrite. This makes movement much easier, as we don't have to deal with branches and can always write about 'moving away from' and 'moving towards' the soma in an unambiguous way. Only in section 5.4, we will explain a separate rule for branching.

Box shaped LSTDP

The first (and simplest) form of Location STDP that we implemented is induced straightaway from regular Weight STDP rules. The concept is that we want synapses to move to those locations where they are optimally potentiated (by Weight STDP, see 5.1.4). For example, if the Weight STDP rule defines that synapses are strengthened with timings between +10 and +20 ms, we would like to move synapses such that they get these timings. If a particular synapse receives a BPAP only 5 ms after it has evoked an EPSP, we want the timing to become larger and we therefore move the synapse away from the soma, vice versa for timings larger than +20 ms. To avoid infinite moving away from the soma with uncorrelated input, we have to make sure that the integral of the window that moves synapses towards the soma (negative on y-axis) is larger that the integral that moves synapses away.

In the simulations in which we used Box shaped LSTDP, the windows for moving away from and towards the soma were [-50, +10] and [+20, +90] respectively, as shown in figure 5.3a.

Hat shaped LSTDP

This second Location STDP rule is almost equal to the Box shaped LSTDP, with the difference that we now define three different 'boxes' instead of two. And with the parameters we used, these three boxes looked like an inversed hat: symmetric, with one large box between two equivalent small boxes (see figure 5.3b). This rule has been observed in real neurons and we applied it as Location STDP

rule, to see to what end it can be used. We will meet this rule again in section 5.3 about correlation detection. Note that it only uses spike timings, just like the previous rule. The basic idea is that synapses that fire correlated with the output of the neuron often have small spike timings and therefore remain close to the soma, while other synapses move away. The window boundaries in our simulations are –70, -30, +30 and +70 ms.

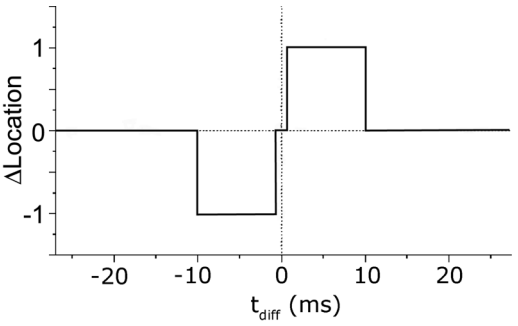

Figure 5.4. Optimal timing LSTDP. Instead of spike timings, t_{diff} is used, incorporating the amplitude of the BPAP as a measure of distance.

Optimal timing LSTDP

This rule is inherently different from the previous two rules, as it uses more information than just spike timing: in an implicit way, the current distance to the soma is included in its operation. Suppose that at some time, the distance from a synapse to the soma is d nodes. If this synapse evokes an EPSP, which travels 1 node/ms towards the soma, it takes d ms for the EPSP to arrive in the soma. Suppose that the synapse contributes to the action potential that is immediately triggered in the soma. The BPAP also takes d ms to arrive in the node where the synapse resides. To summarise: if we assume no preference for absolute location, the optimal spike timing for a synapse that is d nodes away from the soma is $2d$.

As we only want to use local information in our STDP rules to keep them realistic, it isn't possible to specify a rule using the term $2d$. But an action potential always starts with the same amplitude in the soma and if the decay along the dendrite would be approximately the same everywhere, a synapse could use the amplitude of a BPAP as a measure for distance. If we recall that we multiply the BPAP with decay constant κ in each dendrite node of our model, we realise that we can extract the required distance information from the BPAP amplitude.

The goal of the rule is to relatively position the synapses in such a way that they all achieve their 'optimal timing' as often as possible. To achieve this, we compare the actual spike timings, $t_{bpap} - t_{epsp}$, with the optimal values. The difference in spike timings t_{diff} is defined by:

$$t_{diff} = (t_{bpap} - t_{epsp}) - 2d \qquad \text{(Eq. 5.6)}$$

wherein we define d as:

$$d = \log_{\kappa}(BPAP) \qquad \text{(Eq. 5.7)}$$

and $BPAP$ is the amplitude of the BPAP and κ is the BPAP decay constant. If t_{diff} is 0, there is no difference between actual and optimal timing and the synapse is in its currently optimal location. When t_{diff} is above 0, the actual timing is smaller than the optimal timing and we should move the synapse away from the soma; vice versa when t_{diff} is below 0. Naturally, we have to settle lower and upper boundaries for the values of t_{diff} that we deal with, to avoid unwanted in-

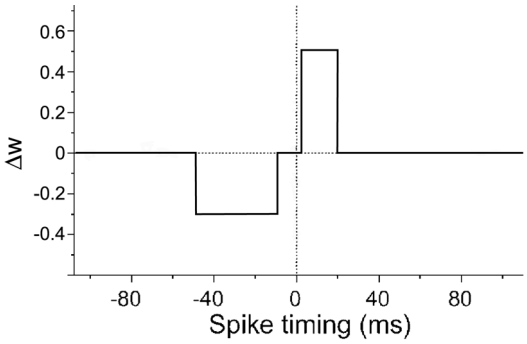

Figure 5.5. Window for both Box shaped WSTDP and BPAP constrained WSTDP. Spike timings between –50 and –10 ms cause LTD, timings between +2 and +20 ms cause LTP. Based on the regular STDP window shown in figure 2.6.

teractions between multiple events (and the requirement for unlimited storage of past events). In our simulations, the windows for moving to and away from the soma were [-10, -1] and [+1,+10] (see figure 5.4).

5.1.4 Weight STDP

Weight STDP (WSTDP) is what we normally consider to be STDP: WSTDP rules use spike timings to modify synaptic weights. In chapter 2, we have already seen that many different rules have been observed in real neurons. We have chosen to implement a few of the most common that seemed to fit well in our model. We also developed a few variations to serve specific functions. In this thesis, we will only describe the two we used in the simulations we present in this chapter. Independent of the type of WSTDP used, initially the weight of synapses is always set to w_{init} = 0.5 (all synapses have equal efficacy to start with).

A difficult issue is whether synaptic weights should be changed in an additive or multiplicative way. With additive STDP, the weight change is a fixed value independent of the current weight, while multiplicative STDP changes the weight dependent on its current value. Lower and upper boundaries have to be implemented in both cases to prevent the weights from increasing/decreasing forever. With additive STDP and a learning window as in figure 2.6, this means that weights converge to a bimodal (min/max) distribution, while weights never get to the boundaries and may converge to more diverse values with multiplicative forms [52,53]. The conclusion is that implementation is easy, but it can be done in many different ways and it is often not obvious what will give the best result in advance.

Box shaped WSTDP

This rule is a simplification of the most common observed rule: positive spike timings between +10 and +20 ms result in LTP, whereas negative spike timings between –50 and –10 ms result in LTD. The experimentally-observed rule (see figure 2.6) has an increasing, saturating curve, but we discarded this and used two boxes instead. The one we used is shown in figure 5.5.

A few remarks should be made. First of all, we enlarged the positive window to contain all spike timings between +2 and +20 ms, because this fits better in our model. Reason is that a synapse at the first dendrite node that contributes to the firing of the soma may have spike timings much smaller than +10: only 2 ms is required for an EPSP to arrive in the soma and a BPAP to get back to the synapse.

Secondly, the peak of LTP is larger than the bottom of LTD. This is done on purpose, to make sure that LTP wins when synapses have just as many LTD as LTP events. Thirdly, the integral of the LTD window is larger than that of the LTP window for reasons of stability: synapses that are uncorrelated with the output of the soma experience more LTD events and get depressed. Lastly, to

avoid unlimited decreasing or increasing weights, weights are constrained between 0 and 1.

Our implementation of the rule is able to operate in two modes: additive and multiplicative. In additive mode, a fixed small value (determined by the LTD/LTP windows) is added to the synaptic weight, no matter the current synaptic weight. In multiplicative mode though, the weight change depends on the current weight. The weight change is multiplied with the current weight w (LTD) or $w_{max} - w$ (LTP), where w_{max} is the maximum weight (1 in this rule). As has been reported before [53], we found that the additive method results often in either full depression or potentiation, while the multiplicative method never gets to the weight boundaries and weights converge to multiple states. However, we experienced that the additive mode was often better suited for our purposes, as the differences between the weights of correlated and uncorrelated synapses are larger. That is why we'll only use additive WSTDP. The LTD window ranged from –50 to –10 ms with value –0.03, the LTP window ranged from +2 to +20 ms with value 0.05.

BPAP constrained WSTDP

To avoid endless growing or shrinking with Weight STDP, we have to introduce boundaries that constrain the weights: with the previous rule, all weights are always between 0 and 1. However, this rises a problem: synapses that are correlated with the output of the soma get potentiated, but they all saturate to w_{max}, which we fixed on 1. Therefore, synapses that are 5 nodes away from the soma may get the same maximum weight as proximal synapses, while their EPSPs decay a lot more before they arrive in the soma. This results in unequal synaptic efficacies: synaptic weights are equal, but because the distances from the soma are different, the EPSPs are of different size when they arrive in the soma.

We would therefore like to include the distance to the soma in the rule. In 5.1.3, we have already done something similar in the Optimal timing LSTDP rule and we can define a dynamic maximum weight using the BPAP amplitude:

$$w_{\max} = \frac{w_{init}}{BPAP} \qquad \text{(Eq. 5.8)}$$

The minimum weight is unchanged: 0. Previously, the size of the weight change was fixed, but now that we have a dynamic maximum it is better to define the weight change as a percentage of the current maximum, for both LTD and LTP. A single event now has the same effect on the synaptic efficacy everywhere, independent of the location of the synapse. The default window is the same as with Box shaped WSTDP, with the difference that the values are percentages instead of absolute values.

5.1.5 STDP example

To give the reader a better idea of how the model and especially the STDP rules behave in a simulation, let us consider an example. Suppose we have a neuron with a soma, four dendrite nodes and three synapses as shown in figure 5.6 at time t = 0 (it is obvious that the simulation has already run before the neuron arrived in this state, but we proceed from here to keep the example short). The synapses are positioned on the first, second and fourth node. We will focus on the synapse on the last dendrite node, which has a weight w of 0.6 at the start of the example.

The synapses will now fire one after each other: the right at t = 0, the middle at t = 1 and the left at t = 2. At t = 3, the EPSPs of the two leftmost synapses have arrived in the soma, the EPSP of the rightmost synapse hasn't. Suppose these EPSPs are enough to trigger an action potential. One time step later, the EPSP of the last synapse arrives: too late to contribute to the somatic AP. Meanwhile, the action potential starts back propagating in the dendrite.

At t = 7, the BPAP arrives in the rightmost synapse and both Location STDP and Weight STDP are applied. Assume we apply Optimal timing LSTDP: the actual timing $t_{bpap} - t_{epsp}$ is 7, the optimal timing is 8 (twice the distance from the soma, which we compute from the amplitude of the BPAP). t_{diff} = -1, which means we move towards the soma one node. No matter which of the two WSTDP rules we described we use, the synaptic strength is increased: the spike timing 7 is between 2 and 20. If we would use Box shaped WSTDP with LTP peak 0.1, this would result in a new synaptic weight of 0.7.

5.1.6 Scaling of the EPSP

A difficult issue is the global EPSP scale that determines the (maximum) size of a single EPSP: one wants the soma to fire regularly, but not all the time. High input rates, more synapses and stronger synaptic weights strongly increase the output rates, which is an undesirable state. On the other hand, reducing their values too strongly will result in elimination of somatic spiking. We therefore sought means for appropriately tuning the EPSP scale to the neurons output rate. We describe three methods, of which we will use only the first: manual.

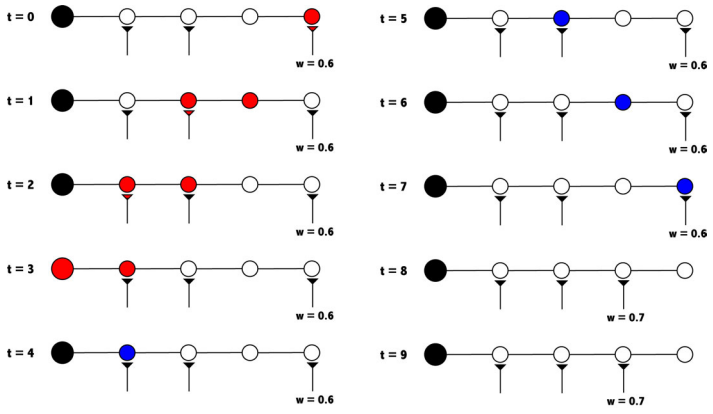

Figure 5.6. Adaptive Neuron STDP example with Optimal timing LSTDP. Three synapses fire in a sequence (synaptic triangles coloured red), the EPSPs are propagated forward through the dendrite (nodes coloured red), an action potential is triggered (soma red) and the AP is back-propagated (nodes blue). At t = 7, Optimal timing LSTDP causes the rightmost synapse to move one node left.

Manual

We set the EPSP scale manually and it was kept constant during each experiment. For each series of experiments, we will list the scale we used. The

chosen scale is based on the number of EPSPs that we thought should cause a somatic AP. We determined empirically what was a good scale. With manual scaling, the scale is independent of the output rate – the output rate is uncontrolled.

Somatic background noise

Another idea was to introduce noise in the membrane potential of the soma, as if there were other synapses randomly firing connected to the neuron. Each time the membrane potential V is updated, we add an additional term V_{noise}:

$$V_{noise} = coin(p_{noise}) * noise \qquad \text{(Eq. 5.9)}$$

where p_{noise} is the probability that noise is added in one time step and *noise* is the amount of noise that is added (in mV). So in a single time step, we add either no noise or the full amount of noise as defined by *noise*.

Although (depending on the parameters) we could get either a slightly depolarised membrane all the time or a more varying noise landscape, it didn't help solving the problem of the EPSP scale. A low p_{noise} has the same effect as lowering the threshold θ or increasing the EPSP scale, while a higher p_{noise} actively contributes to spiking only now and then, but this doesn't help self-organisation as this is purely random.

Rate based EPSP scale

The next possible solution for the EPSP scale problem is more or less equal to having a sliding threshold in the soma [18], but applied in the dendrite and not in the soma. Scaling EPSPs in the soma isn't likely to be realistic, which is why we chose to do something similar in the dendrite. The concept is that we regulate the firing rate of the neuron by scaling the EPSP scale using a feedback mechanism. This is comparable to what Chance et al proposed: gain modulation regulated by overall activity [8]. Doing this in the soma is very straightforward, as this is the place where action potentials are generated. But the dendrites in our model have also access to the firing rate using only local information: each AP results in a BPAP that goes through all dendrite nodes.

The more BPAPs pass through the dendrite, the smaller we want EPSPs to become. If no BPAPs occur, the EPSPs should become larger and larger until an AP is triggered. For this purpose, we define a rate based scale φ which operates in the dendrite node. When rate based EPSP scaling is enabled, equation 5.2 (that defined the integration of charges in a node) is replaced with:

$$Q_t^n = \lambda\left(\sum_{x \in Nodes(n)} Q_{t-1}^x + \sum_{s \in Syn(n)} \frac{q_t^s}{\varphi_t^n}\right) \qquad \text{(Eq. 5.10)}$$

where the rate based scale φ for node n at time t is defined as:

$$\varphi_0^n = 1$$

$$\varphi_t^n = \begin{cases} \varphi_{t-1}^n + c & if\ BPAP_t^n \\ \gamma\varphi_{t-1}^n & otherwise \end{cases} \qquad \text{(Eq. 5.11)}$$

Each node has its own scale value and always starts at 1 (nodes that are created during a simulation start with the parent's scale value). When a BPAP is present in a node, constant c is added: the scale value becomes higher and because all synaptic charges are divided by the scale, synaptic EPSPs are scaled down. When there is no BPAP in a node, the scale is multiplied with a factor γ (<1). Typical values for c and γ are 0.1 and 0.999, respectively. (These values are chosen such that the scale remains approximately 1 at a firing rate of 20 Hz.)

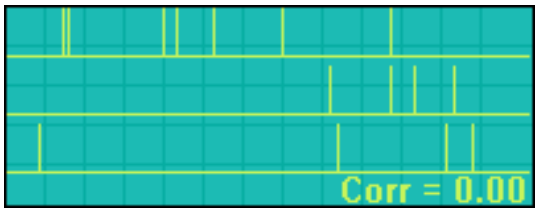

Figure 5.7. Three independent Poisson spike trains at firing rate 20 Hz.

For extreme cases, the rate based EPSP scale works fine: very low firing rates (< 5 Hz) are significantly increased and very high firing rates (> 50 Hz) are decreased. For the 'regular' firing rates between these boundaries though, the rate based scaling only has a small effect; although it works better than the somatic noise, it doesn't solve the EPSP scale problem completely. As we intend to look at the general case and not specific at these boundary cases, we decided to disable it for our experiments. Still, it can be helpful sometimes, especially when the number of synapses is increased (10+).

5.2 Adaptivity with uncorrelated Poisson input

This section describes how the model behaves when we use only uncorrelated spike trains as input. These spike trains are homogenous Poisson processes [10], as all events are statistically independent and have a stationary average firing rate. Firing rates are expressed in Hz: a spike train at 20 Hz contains on average 20 action potentials per second. Because our model is discrete and has time steps of 1 ms, 20 APs in 1000 time steps would equal 20 Hz. To generate a spike train S with rate r, we use:

$$S(t) = coin(\frac{r}{1000}) \qquad \text{(Eq. 5.12)}$$

where $S(t) = 1$ means that a spike occurred at time t and $S(t) = 0$ means that no spike occurred at time t. It is clear that $S(t)$ doesn't depend on t, therefore all events are statistically independent of each other.

As there is nothing to detect in uncorrelated input, our expectation is that the model neuron won't converge to any specific configuration. To allow more room for more interesting simulations later, we will restrict ourselves to a few basic simulations now. In section 5.2.1, simulations with Box shaped LSTDP in combination with BPAP constrained WSTDP are shown, section 5.2.2 shows what Optimal timing LSTDP and BPAP constrained WSTDP do. In both cases, only 3 synapses are used. Example input spike trains for three synapses are shown in figure 5.7. All parameters of the model are set to their defaults, as described in 5.1 (for quick lookup: see Appendix A). The three synapses are initially located on the first and only dendrite node and the EPSP scale is set to 40 (which means that with the initial weights, 3 EPSPs are required to trigger an

action potential). Each run is 10.000 simulated milliseconds, each experiment consists of 10 runs.

5.2.1 Box shaped LSTDP

Figure 5.8a shows an example run with parameters as described above and BPAP constrained WSTDP enabled. In this type of graphs, the soma potential is drawn as +30 mV when an action potential occurs. In the simulations in this section, there is hardly any difference between Box WSTDP and BPAP constrained WSTDP; only the latter is considered. The synapses have an input firing rate of 20 Hz and from the graphs we may conclude that weights are more or less increased and synapses move slowly away from the soma. As the increase in weight doesn't compensate for the increased distance from the soma, the firing rate of the neuron decreases over time. This is due to the decay of the EPSPs in the dendrite. With a higher firing rate of 60 Hz (see figure 5.8b), the synapses move away from the soma much faster and the neuron ceases to fire. With STDP rules, no activity means no plasticity and the neuron 'dies'. Although there is some general tendency in the runs, no structure, such as a preferred location or clustering synapses, emerges in either synaptic location or weight: the changes are random and independent of each other.

When we explained the WSTDP and LSTDP rules, we expected synapses uncorrelated with the output of the neuron to stay close to the soma and get depressed, but this is not what happens here. Because there are only three synapses, the soma fires always as a result of EPSPs generated by one or a few of these three synapses. Therefore, the synapses are *not* uncorrelated with the output of the soma. If we add more synapses with the same type of input (not shown) or add synapses that are more correlated with the output, our expectations become true. We will see examples of this in sections 5.3 and 5.5.

The decay of EPSPs that are propagated forward in the dendrite has a large effect on the behaviour of the neuron. To illustrate this, figure 5.9a shows an example run with three synapses at 20 Hz without Weight STDP. The Location STDP makes the synapses move away from the soma and as the synaptic weights remain the same, the synaptic efficacies decrease and the neuron 'dies'.

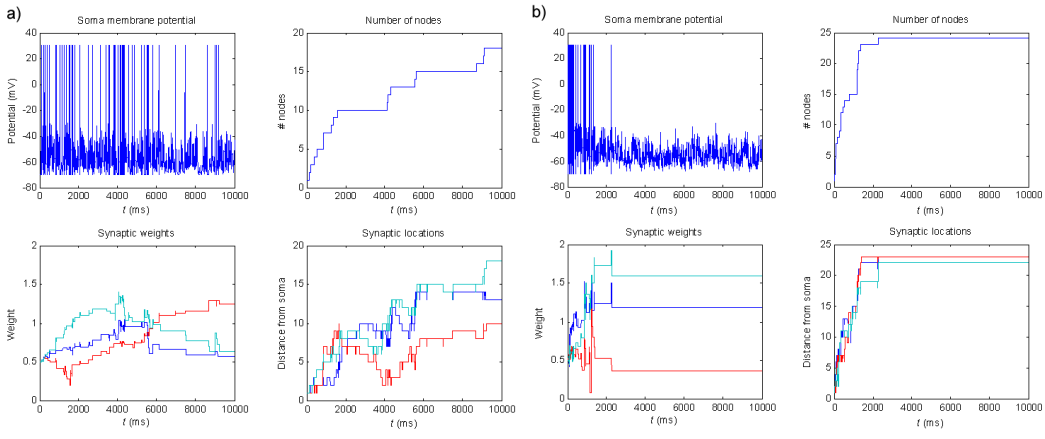

Figure 5.8. Two runs with three synapses receiving three Poisson inputs, Box shaped LSTDP and BPAP constrained WSTDP. Top left: somatic membrane potential during the run; top right: dendrite length during the run; bottom left: synaptic weights during the run; bottom right: synaptic locations during the run. a) Inputs at 20 Hz. b) Inputs at 60 Hz.

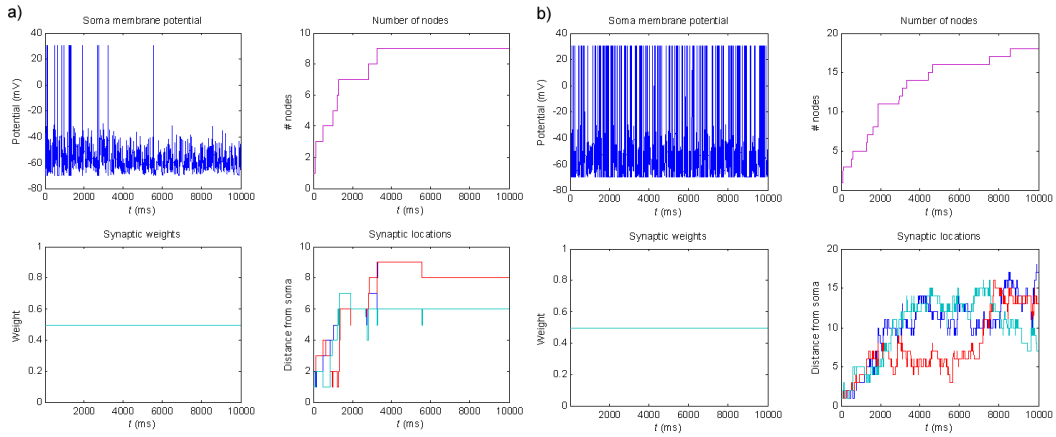

Figure 5.9. Two runs with three synapses receiving three Poisson inputs at 20 Hz, Box shaped LSTDP and no WSTDP. Top left: somatic membrane potential during the run; top right: dendrite length during the run; bottom left: synaptic weights during the run; bottom right: synaptic locations during the run. a) With dendritic decay. b) Without dendritic decay.

On the other hand, if we now disable the forward decay of EPSPs, moving away from the soma no longer affects the synaptic efficacy but only the delay: the firing rate of the neuron remains constant (see figure 5.9b).

Using only Weight STDP (which is more traditional) is also possible: in figure 5.10, an example run in which Location STDP is disabled is shown. The synapses remain at the first node of the dendrite and synaptic weight (and thus synaptic efficacy) are almost constant. In figure 5.11a, the effect of using LSTDP and/or WSTDP with/without dendritic decay on the output rate is shown. When we have no WSTDP and no decay, LSTDP only influences the dendritic delay and output firing rate increases linearly with the input firing rate. When we apply only WSTDP, the output firing rates are about the same up to 40 Hz: at higher rates, more STDP interactions occur and synapses are slightly depressed. Dendritic decay with relocation and no WSTDP results in 'dying' neurons, as we saw before: the output rate is always very low. Our 'normal' situation, with LSTDP, WSTDP and decay, has fairly low firing rates under these circumstances, with optimal synaptic efficacies at a firing rate of 20 Hz.

5.2.2 Optimal timing LSTDP

With three Poisson synapses, it doesn't really matter whether one uses Box shaped LSTDP or Optimal timing LSTDP: there is no pattern to detect in the input and synaptic movement is stochastic. A difference we may conclude from comparing figures 5.11a and 5.11b is that Optimal timing LSTDP tends to move synapses less fast away from

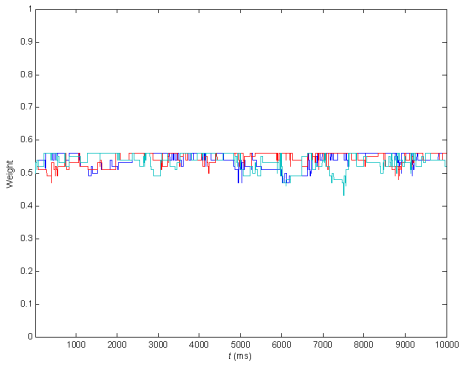

Figure 5.10. Run with three synapses receiving three Poisson inputs at 20 Hz, no LSTDP and BPAP constrained WSTDP. Synaptic weights are more or less kept between 0.5 and 0.6, keeping synaptic efficacies and thus output firing rate constant.

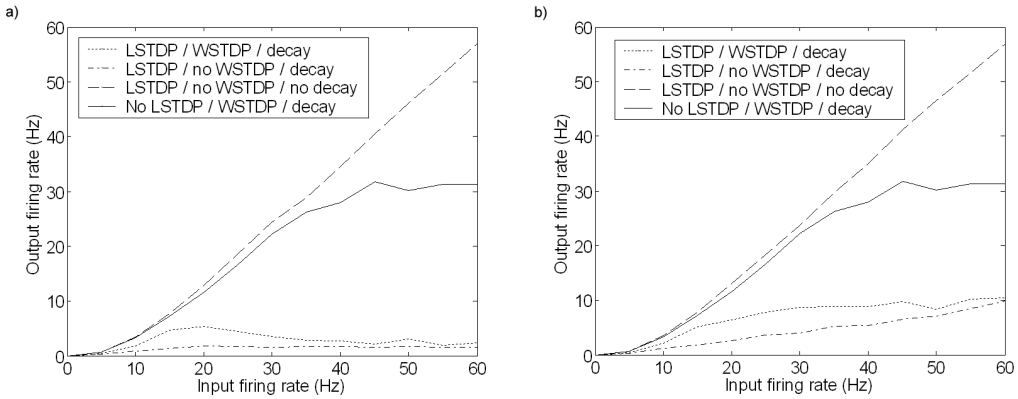

Figure 5.11. Effect of input firing rate on output firing rate with three synapses receiving independent Poisson inputs and BPAP constrained WSTDP. The effect of four combinations of applying LSTDP, WSTDP and dendritic decay are shown. Average over 10 runs. a) Box shaped LSTDP. b) Optimal timing LSTDP.

the soma with the current input: this causes the higher output firing rates in two cases and is also visible in figure 5.12.

5.3 Sequence detection

In the previous section we observed how the neuron develops given uncorrelated Poisson input. In this section, we'll explore how STDP rules may develop neural structures capable of detecting temporal sequence. The input spike trains containing the sequences are generated using what we called the Skewed Single Interaction Method (SSIP), based on the Single Interaction Method (SIP) by Kuhn et al. [29]. In this SIP, a single interaction Poisson process $w_u(t)$ (with rate α) is duplicated to multiple independent Poisson processes $w_i(t)$ (each with rate β), resulting in Poisson spike trains that have a certain overlap. In our SSIP method, the interaction is no longer simply duplicated to all spike trains, but with a certain offset o with respect to the previous instance. This way, the synapses fire in a certain sequence, as illustrated in figure 5.13a. When the count correlation coefficient c is 1, only the interaction process is used and the synapses always fire in sequence. However, when we lower the correlation, independent events also occur in the individual spike trains. To vary the correlation c while keeping the total firing rate r steady, we compute the interaction firing rate α and independent firing rate β as follows:

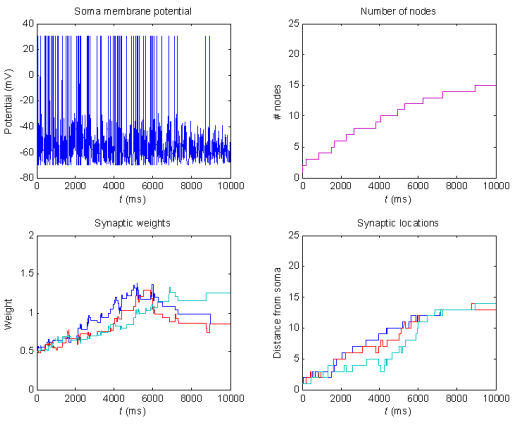

Figure 5.12. Run with three synapses receiving three Poisson inputs at 60 Hz, Optimal timing LSTDP and BPAP constrained WSTDP. Synapses move away from the soma slower than with Box shaped LSTDP, but in the end the result is the same: the synapses are too far away from the soma to trigger action potentials.

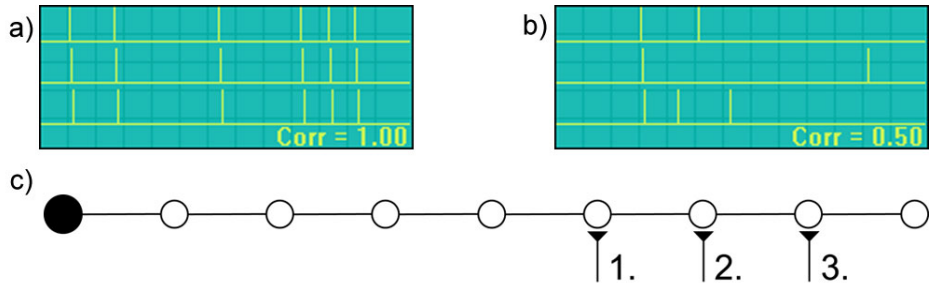

Figure 5.13. a) Three Skewed SIP generated spike trains with correlation 1.0. b) Three Skewed SIP generated spike trains with correlation 0.5. c) Synapses ordered on the linear dendrite based on the sequence in the input; dispersed order.

$$\alpha = cr$$
$$\beta = (1-c)r \qquad \text{(Eq. 5.13)}$$

Each process $x_i(t)$ in a set of N spike trains with firing rate r and correlation c is now defined as:

$$x_i(t) = w_u(t - io) + w_i(t)$$
$$(i = 0,1,2,\ldots,N-1) \qquad \text{(Eq. 5.14)}$$

in which o is the offset. Examples of three spike trains generated with this method are shown in figures 5.13a and 5.13b. The task is to develop a neuron that can efficiently detect temporal sequences by means of STDP rules. When three synapses receive Skewed SIP as input with offset 1 (which is 1 ms in our model), the optimal synaptic configuration would be 3 – 2 – 1: suppose that at time t = 0, synapse 1 evokes an EPSP. This means that one millisecond later, synapse 2 will do the same. Another millisecond and synapse 3 will follow. If these synapses are positioned 3 – 2 – 1 on the dendrite, all EPSPs arrive in the soma simultaneously and thus maximising the probability of evoking a somatic AP. Figure 5.13c shows the reverse of the optimal configuration, the reason for this will soon become clear. The default value for both the SSIP count correlation coefficient c and the offset o is 1.

As measure for performance of this particular task, we decided on a very simple but strict function: the percentage of time steps that the synapses are in their correct order. Note that order means that only relative position matters: the absolute location on the dendrite has no influence on performance. The complete run (10,000 time steps) was taken into account and to ensure that convergence speed did not determine the outcome, we checked this was fast enough (within 2,000 ms). The number of spike timings/STDP events naturally depends on the firing rates, but more events doesn't necessarily result in better performance. Also, we left out standard deviations in the results that follow. There is always variance in performance, but this is due to the stochastic nature of the Poisson spike trains and therefore equal for all experiments. The variance is not always the same, but we found no big differences that are worth mentioning or making the graphs overly complex.

In the previous subsection, we used four combinations of Location STDP, Weight STDP and dendritic decay, but not all combinations are realistic and useful. Also, we restricted ourselves to only one WSTDP rule and we want to extend that in this subsection. Therefore, we will only look at two scenario's here: 1) LSTDP with WSTDP and decay and 2) LSTDP without WSTDP, without decay.

We are investigating LSTDP rules in the first place, which explains why both scenario's include LSTDP. The first case is realistic, as it includes dendritic decay and Weight STDP rules can compensate for this. If we are interested in only the effect of the Location STDP, leaving out the WSTDP rule is the obvious thing to do. But with dendritic decay enabled, we then soon have the problem that the efficacies of synapses that are moved away are reduced to almost zero. Therefore we chose to have implicit synaptic scaling in the dendritic tree in this case: by disabling both WSTDP and dendritic decay, we equalize all synaptic efficacies, no matter where they're located.

5.3.1 Box shaped LSTDP

The Box shaped STDP rule, as deduced in section 5.1.3, orders synapses 1 – 2 – 3 as illustrated in figure 5.13c. This structure reflects a maximal dispersion of the correlated synaptic inputs arriving at the soma, which is why we will call this the dispersed order. Note that in all figures in this subsection, the performance is measured as the percentage of time steps the synapses are in dispersed order, not in the compact order. Although the dispersed order is not the optimal (compact) order, it is the only order we can obtain with Box shaped LSTDP and we will therefore base performance on this. Simulations were performed with 3 synapses receiving SSIP input. In all these simulations, the EPSP scale is set to 40.

Figure 5.14 shows an example run with Box shaped LSTDP and BPAP constrained WSTDP. Blue represents synapse 1, red and green represent synapses 2 and 3, respectively. Synapse 1 remains closest to the soma and it is clear that although they keep on moving, there are large periods in which they're ordered 1 – 2 – 3. The BPAP constrained WSTDP rule is well able to scale the synapses with respect to each other, resulting in unequal synaptic weights but equal synaptic efficacies.

To explain the behaviour of the synapses, we need to take a closer look at the Location STDP rule we are using: what does it do? Initially, the synapses are very close to the soma and whenever the three of them fire EPSPs in a sequence, the soma responds with an action potential. The time for an EPSP to travel to the soma and a BPAP to travel back to the synapse is only 2 ms: the spike timings are very small in the beginning. If we look at figure 5.3a, we see that small spike timings result in move-

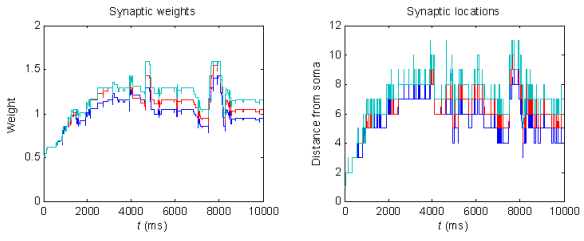

Figure 5.14. Synaptic weights and locations during a run with three synapses receiving SSIP input at 20 Hz, Box shaped LSTDP and BPAP constrained WSTDP. During large parts of the run, synapses are in the dispersed order (right). Furthermore, synaptic scaling is achieved: the weight of each of the synapses (left) is strongly correlated with synaptic locations (distance from the soma).

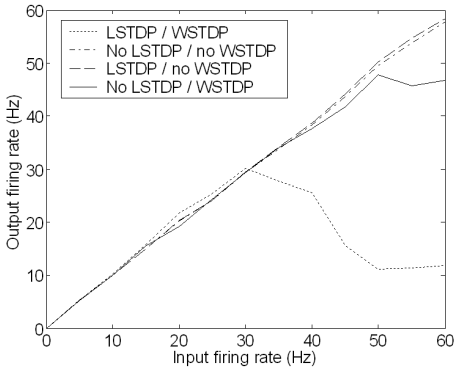

Figure 5.15. Effect of input firing rate on output firing rate with three synapses receiving SSIP input, Box shaped LSTDP and BPAP constrained WSTDP. The effect of four combinations of applying LSTDP, WSTDP and dendritic decay are shown. Average over 10 runs.

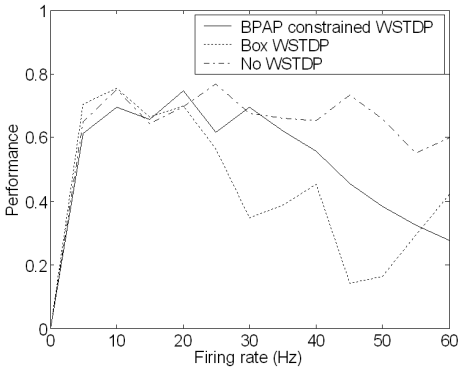

Figure 5.16. Effect of input firing rate on dispersed order performance with three synapses receiving SSIP input and Box shaped LSTDP. In addition, the effect of applying no WSTDP, BPAP constrained WSTDP and Box shaped WSTDP is shown. Average over 10 runs.

ment away from the soma and that's what happens. All three synapses start moving away from the soma and they keep moving away until their spike timings are larger than 10 ms. But there is a difference between the synapses: because they fire in a sequence, the spike timings vary a bit and the synapse that fires last (and eventually triggers the action potential) has to move further away to satisfy the 10 ms than the synapse that fires first. Having read this, one may correctly conclude that the absolute location of the synapses can be manipulated by modifying the boundaries of the STDP window.

With uncorrelated input, 3 synapses firing (almost) at the same time was seldom, but this is different now. As the correlation and offset are both 1, the three synapses always fire together, within 3 ms. This also shows in the relationship between input and output firing rates (see figure 5.15). The Weight STDP causes depression with higher firing rates (> 30 Hz), otherwise the output firing rate is almost equal to the input firing rate. The effect of the (input) firing rate on performance is shown in figure 5.16. With firing rates > 5 Hz, performance is pretty good and only affected by LTD interactions when the rate is higher than 30 Hz. Synapses controlled by Box shaped WSTDP are easier affected than those controlled by BPAP constrained WSTDP, which is not surprising: at higher rates, the probability of being moved away further from the soma is higher and Box shaped WSTDP cannot achieve equivalent synaptic efficacies in that case. Best performance is around 0.75, which means that synapses are ordered dispersed in 75% of all time steps – not a bad score!

Performance within the boundaries of realistic firing rates (between 5 and 50 Hz) is quite acceptable (except for the Box shaped WSTDP case maybe), but what happens when we look at more realistic synapses and input properties? Decreasing the SSIP correlation has a dramatic impact on performance (see figure 5.17a): performance is quite bad with any correlation lower than 1, especially since sequences occurring in real neurons are not likely to be perfect. Also, real synapses are not very reliable and an incoming spike may only result in an EPSP in 50 – 60 % of the cases. Stochastic synapses do exactly that: dependent on release probability (which is the same for all synapses), it is deter-

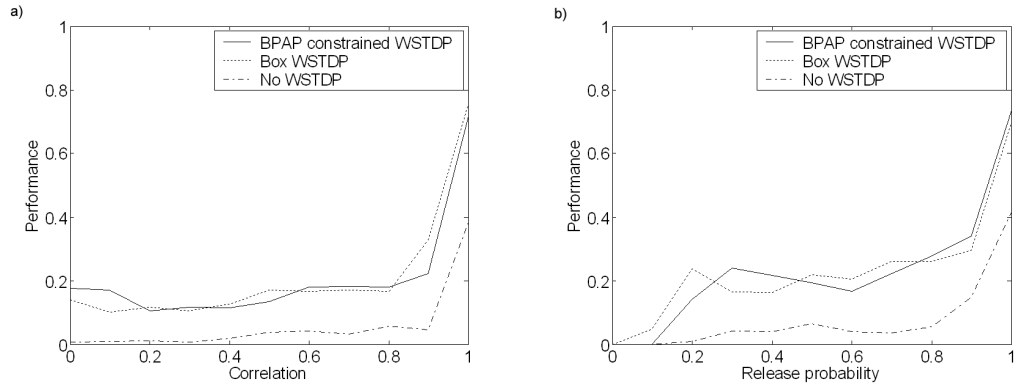

Figure 5.17. Effect of input correlation and synaptic release probability on dispersed order performance with three synapses receiving SSIP input and Box shaped LSTDP. In addition, the effect of applying no WSTDP, BPAP constrained WSTDP and Box shaped WSTDP is shown. Average over 10 runs, input rates 20 Hz. a) Varying SSIP correlation. b) Varying synaptic release probability.

mined stochastically whether an incoming spike causes an EPSP or not in each synapse. In our simulations, the release probability of the synapses are normally 1, decreasing it has almost the same effect as decreasing the correlation (see figure 5.17b). (Both experiments were done at a firing rate of 20 Hz.)

Another type of noise may be introduced by adding more synapses to the model. In these simulations, the three correlated SSIP synapses remain (and performance is measured using their relative order), but N synapses receiving N independent Poisson synapses are added. We call these additional synapses 'uncorrelated', because they fire independent of each other and they are unlikely to become correlated with the output of the soma, especially because the three SSIP synapses always fire together. Figure 5.18a shows a 3D graph in which performance is shown along the z-axis and the other axes represent the release probability of *all* synapses and the number of uncorrelated synapses that were added to the model. Again we see that release probabilities lower than

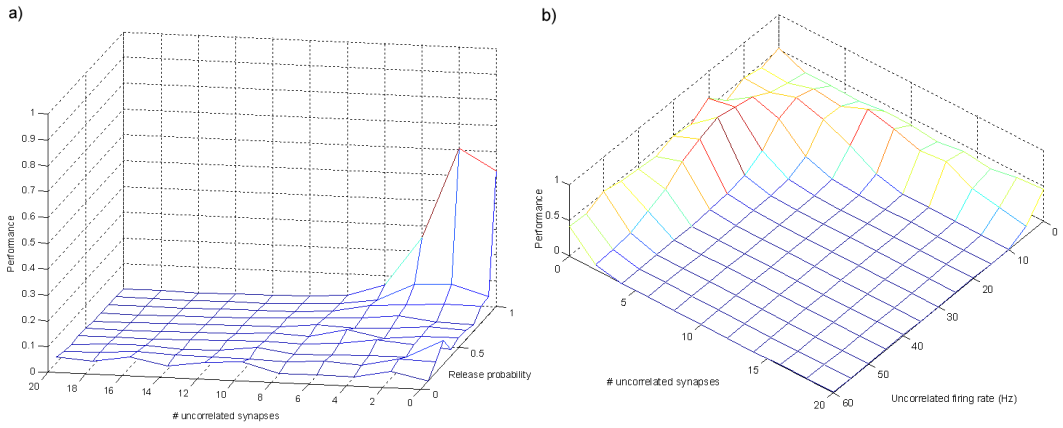

Figure 5.18. Effect of release probability and adding uncorrelated synapses receiving independent Poisson spike trains on performance. Performance measured of three correlated synapses receiving SSIP input at 20 Hz. Average over 10 runs. a) Varying release probability and number of uncorrelated synapses. Poisson rate of uncorrelated synapses is 20 Hz. b) Varying number of uncorrelated synapses and their firing rate. Release probability 1.0.

one result in very bad performance and adding too many uncorrelated synapses (> 4) also disturbs the order of the correlated synapses. This is without WSTDP and decay, but results are almost equal with BPAP constrained WSTDP and decay enabled.

If we look very carefully at figure 5.18a, we may conclude that some 'background' noise (2 synapses) can be helpful to achieve higher performance. This is confirmed by figure 5.18b, which shows that some amount of uncorrelated 'noise' results in a better performance. But too much uncorrelated EPSPs (either too much synapses or too high firing rates) cause distortion of the synapse order. Reason that the noise helps to get a higher performance is that the rule itself needs some noise to achieve the dispersed order: this can be either negative STDP interactions, changing (increasing) weights or adding uncorrelated EPSPs. As we have no WSTDP in this case, some background noise increases the probability that the dispersed order is established, whereas too much noise results in distortion. With BPAP constrained WSTDP, the effect of adding uncorrelated synapses is quite different (see figure 5.19). If we have 3 correlated, 20 uncorrelated synapses and the uncorrelated synapses firing at low firing rates, performance is very bad: the LSTDP rule is unable to maintain the required synapse order. On the other hand, when we increase the uncorrelated firing rate, performance becomes very good: after giving some noise initially, all uncorrelated synapses are depressed and no longer get the opportunity to affect the neuron, whereas the correlated synapses are strengthened (a nice example of the filtering effect that a Weight STDP rule may have). However, the noise in the beginning increases the probability that the three correlated synapses end up in their correct order. Higher correlated firing rates have the same effect as they had before: at rates above 30 Hz, more negative interactions occur and correlated synapses are depressed, resulting in lower performance.

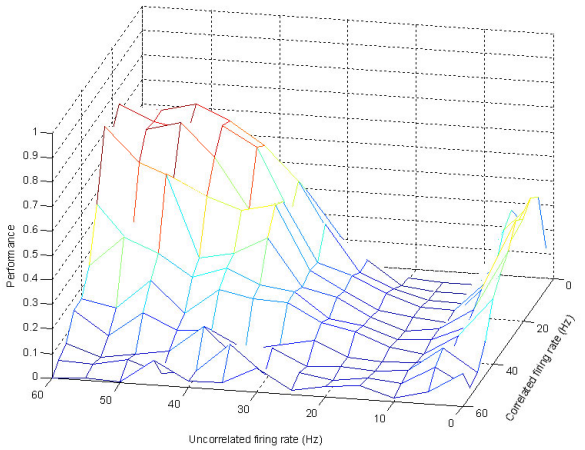

Figure 5.19. Varying the firing rates of correlated and uncorrelated synapses. Performance measured of three correlated synapses receiving SSIP input. In addition, twenty synapses receiving independent Poisson spike trains were added. Average over 10 runs.

5.3.2 Slowing down growth and movement

Dendritic growth and synaptic relocation is very fast: a new dendrite node is added as soon as one synapse located on the last node wants to move and a synapse may move each time step. In other words, the dendrite may grow 1 node/ms and a synapse may move 1 node/ms and thus increase the delay between EPSP generation and arrival in the soma at a very high rate.

Especially when synapses have input spike trains (or periods in these trains) and a LSTDP rule that make the synapse walk away from the soma at a steady pace, it may be useful to slow down both growth and movement. But intuition

tells us slowing down might also be useful in other cases: if there is some noise in the events, do we want synapses to move away from their locations immediately? Even if they return to the same location later?

Slowing down growth and movement can be achieved by introducing thresholds for both. Instead of adding an additional node each time a synapse wants to move to a node that isn't there, we can say that an additional node is created only when X attempts to move there have been made. It doesn't matter if it's the same synapse or not, as long as X requests to add the new synapse have been made. If X equals 1, we are in the same situation as we were before, but increasing X to 2 or 3 decreases the growth speed. In practice, a growth threshold definitely slows down growth, but it doesn't result in better synaptic configurations with the tasks we perform. As we prefer our model to converge to a stable state as fast as possible to reduce the required running times, we didn't use the growth threshold.

The threshold for synaptic movement works slightly different, as a synapse may move in two directions. Normally, Location STDP immediately applies any relocation after evaluation of all spike timings in a single network update. When the threshold is enabled, evaluation of a spike timing results in a relocation value (-1 or +1 for moving towards, respectively away) that is added to the current relocation state of the synapse. This state starts at 0 and is maintained over multiple network updates. As soon as the state crosses a positive threshold Y or negative threshold Z, the synapse is moved either away from or towards the soma and the state is reset to 0. When Y = Z, the synapse has no preference for moving in either location and the direction of movement is only determined by the spike timings and LSTDP rule. When movement of a synapse with thresholds Y = Z = 1 is completely random, increasing these thresholds to higher values may 'stick' a synapse to its position, while synapses that aren't moving at random may still move appropriately (although at a slower pace).

This method works quite well, but as all simulations we will describe are with some sort of Poisson spike trains and thus based on chance, having higher thresholds (Y = Z = 2 or Y = Z = 3) didn't result in better performance. Because of the nature of the inputs, the variance in performance became higher (and some individual runs seemed to do much better), but the average remained the same. And like before, convergence is obviously slower with a threshold enabled, reason enough to disable the feature in our simulations.

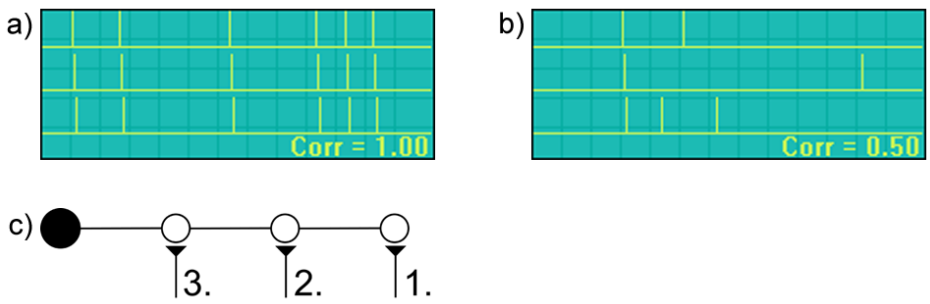

Figure 5.20. a) Three Skewed SIP generated spike trains with correlation 1.0. b) Three Skewed SIP generated spike trains with correlation 0.5. c) Synapses ordered on the linear dendrite based on the sequence in the input; compact order.

5.3.3 Optimal timing LSTDP

We next explored whether any given set of STDP rules can produce a dendrite structure that is capable of obtaining a compact synapse order. This task was only successfully fulfilled by Optimal timing LSTDP. Given the same initial conditions as described in 5.3.1, the synapses were arranged in their compact order 1 – 2 – 3 (see figure 5.20c).

An example run with three synapses receiving SSIP input at 20 Hz is shown in figure 5.21. The BPAP constrained WSTDP controls the synaptic weight the same way as it did before: the synaptic weights are scaled with respect to the distance from the soma and because they are always in the same relative order, the weights are always scaled such that the synaptic efficacies are virtually the same. Not only with respect to each other, but also in an absolute sense: when the synapses move further away from the soma, their weights increase and vice versa when they move back.

The relative order can be established and maintained only because each of the synapses wants to achieve 'optimal' spike timing, which depends on its distance from the soma and the sequence in the input.

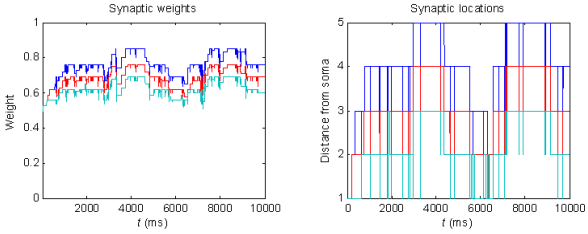

Figure 5.21. Synaptic weights and locations during a run with three synapses receiving SSIP input at 20 Hz, Optimal timing LSTDP and BPAP constrained WSTDP. During large parts of the run, synapses are in the compact order (right). Furthermore, synaptic scaling is achieved: the weight of each of the synapses (left) is strongly correlated with synaptic locations (distance from the soma).

Three correlated synapses

In these experiments, three synapses receiving SSIP spike trains are required to arrange themselves in the compact order using the Optimal timing LSTDP rule. The EPSP scale in these experiments is always 40. The influence of input firing rate on output firing rate is comparable with the one we saw earlier, with the biggest difference that the combination of LSTDP and WSTDP gives higher output rates at higher input rates (see figure 5.22). This can be explained with the

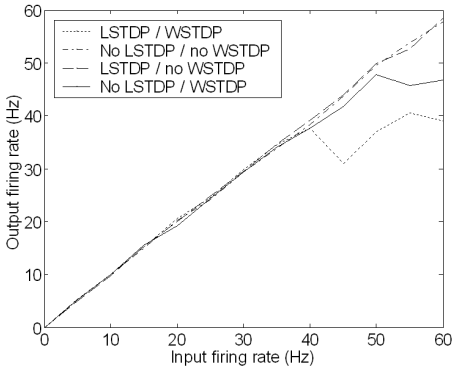

Figure 5.22. Effect of input firing rate on output firing rate with three synapses receiving SSIP input, Optimal timing LSTDP and BPAP constrained WSTDP. The effect of four combinations of applying LSTDP and WSTDP are shown. Average over 10 runs.

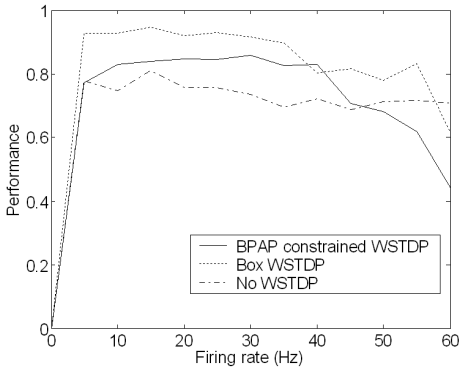

Figure 5.23. Effect of input firing rate on compact order performance with three synapses receiving SSIP input and Optimal timing LSTDP. In addition, the effect of applying no WSTDP, BPAP constrained WSTDP and Box shaped WSTDP is shown. Average over 10 runs.

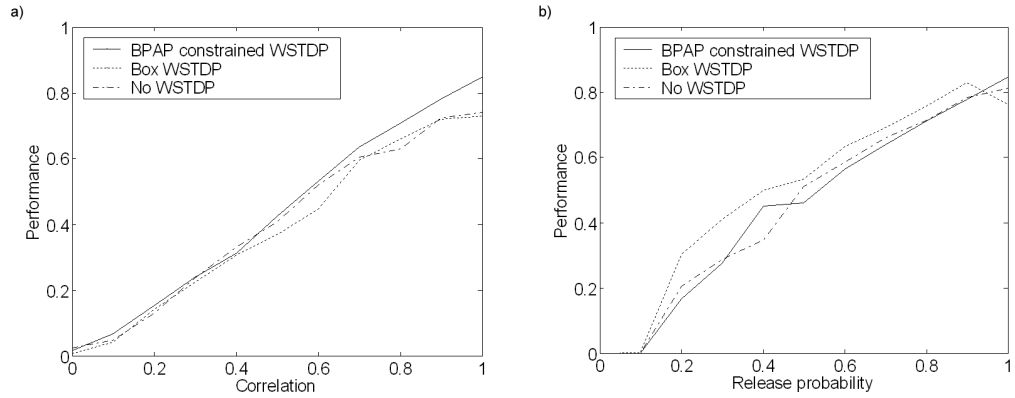

Figure 5.24. Effect of correlation and release probability on dispersed order performance with three synapses receiving SSIP input and Optimal timing LSTDP. In addition, the effect of applying no WSTDP, BPAP constrained WSTDP and Box shaped WSTDP is shown. Average over 10 runs. a) Varying SSIP input correlation. b) Varying synaptic release probability.

fact that this rule moves synapses less far away from the soma, certainly when they are correlated: the rule has no preference for absolute location and keeps the synapses close to their starting positions, unlike the previous rule. Because synapses remain closer to the soma, it is easier for BPAP constrained WSTDP to keep the synaptic efficacies equal.

Performance is pretty good for all firing rates (see figure 5.23), but higher rates in combination with a WSTDP rule induces some LTD and this has a negative effect on performance. Synaptic strength has an influence on performance: without WSTDP (w = 0.5), performance is lower than with Box shaped WSTDP (w_{max} = 1.0). And because the synapses stay close to the soma on average, 0.5 < w_{max} < 1.0 is valid for BPAP constrained WSTDP and performance for this rule is between the other two.

Depending on the WSTDP rule used, maximum performance lies between 0.75 and 0.95. Why never 1.0? This is not only because all time steps are taken into account (including the convergence phase in the beginning), but mainly because there are always occasions in which multiple sequence events occur in a very short period of time (this is simply a property of Poisson processes). When this happens, interactions between the different events may occur and synapse order may be disturbed for a little while, until it is corrected through the next spike timings.

Contrary to what we saw with the Box shaped LSTDP rule, the current rule is more reliable over a larger range of input rates. Although performance degrades with decreasing input correlation (as with Box shaped STDP), it does so less steeply. For all WSTDP scenario's, performance is more or less linear with the correlation of the input spike trains (see figure 5.24a). Using stochastic synapses results in an even more interesting graph: there is a threshold for release probability to achieve any performance, but performance is super linear when this threshold is exceeded (see figure 5.24b). With Box WSTDP, performance is 0.3 with a release probability of 0.2 and 0.5 with a release probability of 0.4. Box shaped WSTDP clearly has an advantage compared to the other two scenario's: as long as the synapses are close to the soma, it can increase synaptic efficacies to a higher level than they are in the other two cases. The maximum weight is 1, while the weight is 0.5 (or slightly above) in the other two cases. In the initial configuration, 3 EPSPs are required to invoke a spike, but

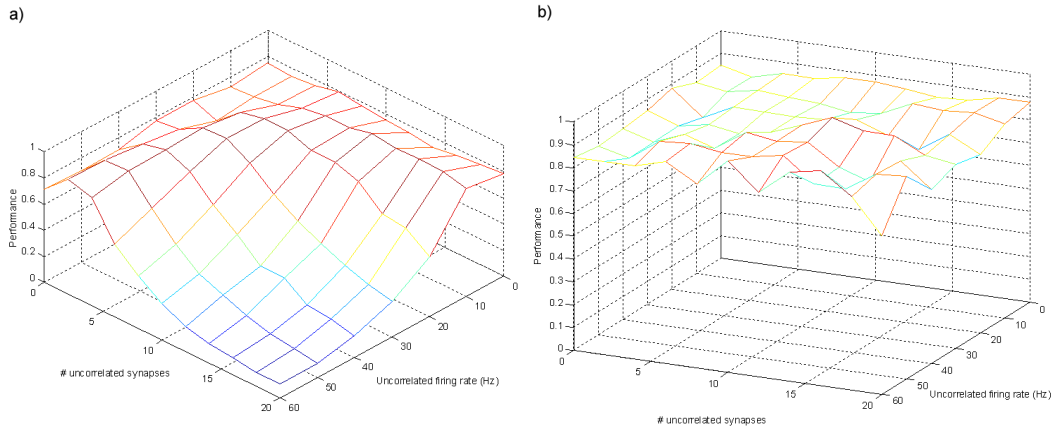

Figure 5.25. Varying the number of uncorrelated synapses and their firing rates. Performance measured of three correlated synapses receiving SSIP input at 20 Hz. Average over 10 runs. a) No WSTDP, no dendritic decay. b) BPAP constrained WSTDP, dendritic decay.

Box shaped WSTDP is able to increase the somatic EPSP amplitudes such that 2 EPSPs are enough to do the same. And this, of course, is an advantage with lower release probabilities, when the probability that all three synapses succeed in firing an EPSP becomes very low.

We now add uncorrelated synapses to provide background noise, just like before. Figure 5.25a shows correct synapse order of three synapses receiving SSIP input at 20 Hz when we add N synapses firing at rate r and assume implicit synaptic scaling by disabling WSTDP and dendritic decay. Just like we saw before, adding some noise has a positive effect on performance, but adding too much uncorrelated firing activity ruins the order of the correlated synapses. How different is this when we use BPAP constrained WSTDP and decay: no matter how many uncorrelated synapses firing at any rate, performance remains good (see figure 5.25b). There is some variance due to the fact that these values are averages of 'only' 10 runs of 10.000 ms, but there is nevertheless a small trend noticeable: increasing the number of uncorrelated synapses and their rates results in even better performance, as they are depressed more effectively. To illustrate this, figure 5.26 shows the spike triggered

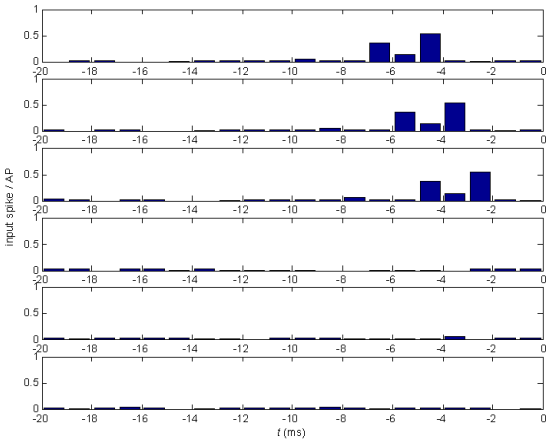

Figure 5.26. Spike triggered average input of the last 5,000 ms of a run with Optimal timing LSTDP and BPAP constrained WSTDP, shown for 6 synapses. The three synapses shown in the upper three graphs received SSIP input at 20 Hz. 20 other synapses received independent Poisson trains at 20 Hz. Of these 20, three were randomly selected and their spike triggered average input graphs are shown at the bottom. For each synapse, the average presynaptic input just before each postsynaptic output is shown for 20 ms before each somatic AP. The 3 correlated synapses almost always fired 2 to 7 ms before an action potential was generated, while the other synapses seem completely uncorrelated with the output of the neuron.

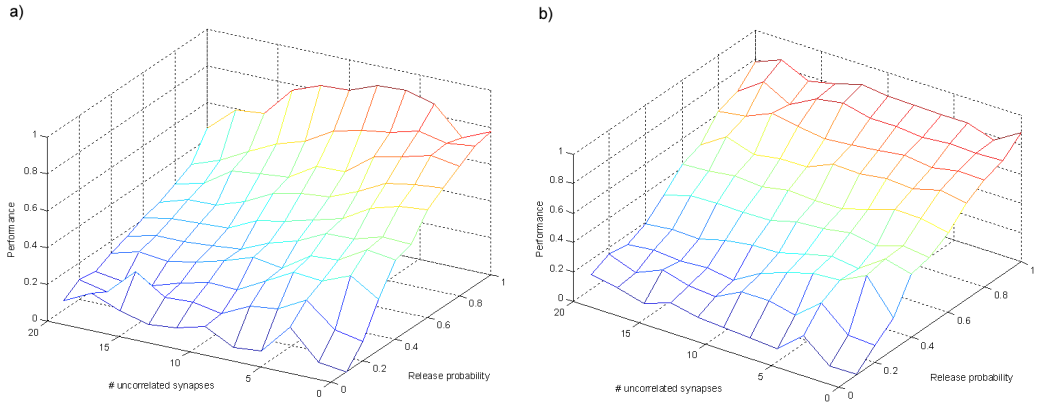

Figure 5.27. Varying the number of uncorrelated synapses and synaptic release probability. Performance measured of three correlated synapses receiving SSIP input at 20 Hz, uncorrelated synapses also received input at 20 Hz. Average over 10 runs. a) No WSTDP, no dendritic decay. b) BPAP constrained WSTDP, dendritic decay.

average input of 3 correlated and 3 (out of 20) uncorrelated synapses. Summed over all output spikes of the neuron, these graphs show the history of incoming spikes just before the output spikes per synapse. It is clear that the three correlated synapses received a spike just before an output spike in virtually all cases, while there is no such distinct relation visible in the graphs of the uncorrelated synapses. Even the sequence of the three correlated synapses can be deduced from this figure.

When adding uncorrelated synapses, the influence of using stochastic synapses with varying release probabilities doesn't change (see figure 5.27a). Without WSTDP, the negative effect of adding more uncorrelated synapses is visible, but when we use WSTDP, the effect of varying release probabilities is really independent of the number of uncorrelated synapses (see figure 5.27b).

Ten correlated synapses

Optimal timing LSTDP with three synapses performed well under all conditions tried so far. When we increase the number of correlated synapses to 10, we also have to decrease the EPSP scale. Otherwise, we would stick with the unrealistic regime in which only 3 EPSPs are enough to trigger an action potential. In the scenario with BPAP constrained WSTDP, we decided on an EPSP scale of 10 (initially requiring all 10 correlated synapses to fire), but for the scenario without WSTDP and dendritic decay we chose 12 as EPSP scale. Reason is that there is no change in synaptic efficacy possible here, while BPAP constrained

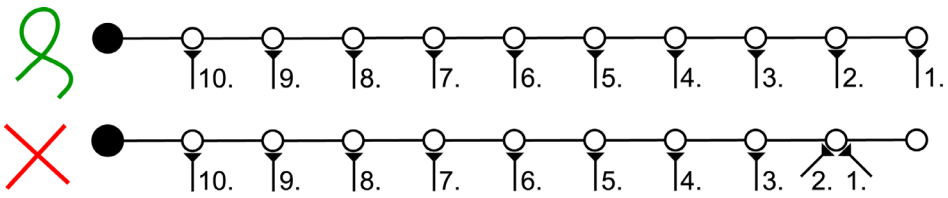

Figure 5.28. Performance measure for sequence detection with 10 synapses. Synaptic order is only correct when all 10 synapses are arranged as shown in the top illustration (absolute position doesn't matter). As soon as one synapse is out of position, performance is zero (bottom).

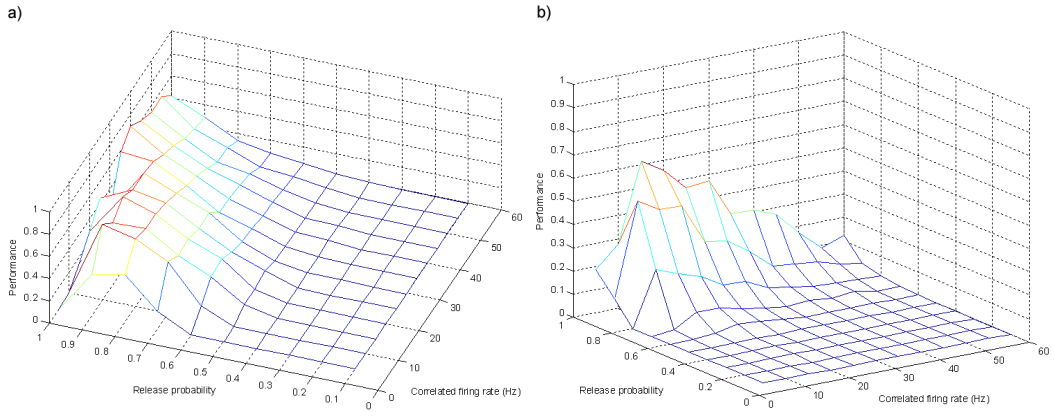

Figure 5.29. Varying synaptic release probability and input firing rate with 10 synapses. Performance measured of 10 correlated synapses receiving SSIP input. Average over 10 runs. a) No WSTDP, no dendritic decay. b) BPAP constrained WSTDP, dendritic decay.

WSTDP can cause a slight increase in efficacy compared to the starting configuration.

In the three synapses situation, it is easy to choose an appropriate EPSP scale, but this becomes more difficult with ten synapses. If the EPSP scale is too small, the initial configuration in which all synapses are located on the same node causes a spread in arrival of EPSPs in the soma and no action potential is triggered. On the other hand, if the EPSP scale is too large, only a few EPSPs cause an action potential and the synapses no longer form a single sequence. Also, the dendrite is not allowed to grow in the direction of the soma, while synapses try to move there in this situation. The rate based EPSP scaling described earlier in combination with a very small initial EPSP scale partly solves the problem, but also introduces problems, like extremely slow convergence. We instead chose the EPSP scale manually, such that the initial configuration evoked somatic activity and plasticity was possible.

We use the same performance measure as before, but we should realise this is a very strict measure! Previously, only three synapses had to be ordered correctly, whereas now all ten synapses have to be in the correct position to yield a positive performance (see figure 5.28). As the probability for a single synapse to be 'disturbed' for a short time is the same and many more synapses exist, it is impossible to compare absolute performance in the following with what we saw earlier.

The largest problems arise when synaptic efficacies are completely static: figure 5.29a shows that a release probability of 1 leads to a very bad performance for firing rates below 35 Hz, but pretty good above. Higher rates cause enough action potentials to make it possible to move synapses away from their initial position (which often seemed the biggest problem). With lower rates and 'perfect' synapses, action potentials are invoked before the last synapse in the sequence has fired and this last synapse is left out of the sequence. Lower release probabilities help to get this last synapse into the sequence, as not all synapses always fire and the last synapse may be very important to help trigger an action potential. Despite these scaling problems, a best performance of 0.7 was achieved for $p_{release} > 0.9$ at all frequencies. With WSTDP, performance is not as good: best performance at 15 Hz is around 0.6 (see figure 5.29b). With this larger number of synapses in a sequence (and larger distances from the

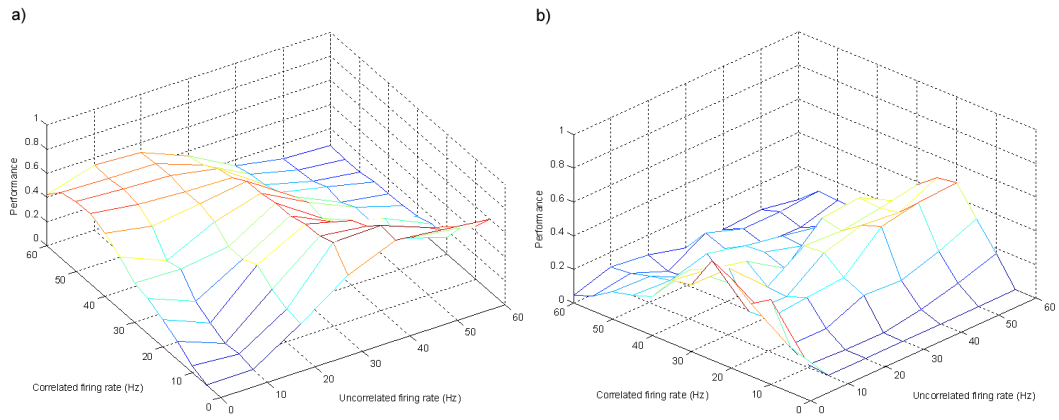

Figure 5.30. Varying uncorrelated and correlated firing rates with 20 uncorrelated and 10 correlated synapses. Performance measured of 10 correlated synapses receiving SSIP input. Average over 10 runs. a) No WSTDP, no dendritic decay. b) BPAP constrained WSTDP, dendritic decay.

soma!), the Weight STDP seems to be unable to keep the synaptic efficacies equal. One reason for this is that the WSTDP LTP window only ranges up to +20 ms, while a synapse that is more than 10 nodes away from the soma cannot achieve this timing. Especially at higher rates, LTD seems to win over LTP and degrades the performance. This is not surprising though, as synapses may move one node per spike timing, while synaptic weights are modified much slower.

The observation that noise is required to achieve good performance (in the scenario without WSTDP) is confirmed when uncorrelated synapses are added. In figure 5.30a, it is shown that adding 20 uncorrelated synapses firing at high rates improves the performance (up to 0.7). When BPAP constrained WSTDP is enabled, adding noise has a negative influence on performance: only when the 20 uncorrelated synapses fire at high rates and are effectively filtered out by LTD, performance is back at the same level as without noise (see figure 5.30b).

Varying the correlation lag

Until now, we have only varied the number of synapses, input firing rates, WSTDP scenario's and types and amounts of noise: input correlation, release probabilities and additional uncorrelated synapses. As mentioned in the description of the model, there are many more parameters one could investigate and we have actually done a lot more, but we have selected the results that we think are most interesting for this thesis. We will now shortly discuss two more parameters that we think are worth considering here: the Optimal timing LSTDP window and SSIP offset.

In the simulations described in this thesis, we used a fixed LSTDP window, as shown in figure 5.4. The window is fully symmetric, as this minimizes the possibility that synapses 'walk' either away from or towards the soma, disregarding the particular input. The 'inner' boundaries –1 and +1 could be changed, but this would mean allowing 'imperfect' synapse orders: when these boundaries are set to –2/+2, a difference of 1 ms between optimal timing and actual spike timing is allowed. As we are interested only in 'perfect' ordering here, we used only –1/+1.

The outer boundaries define the maximum difference between actual and optimal timing that is corrected for. For example, if we would have the default boundaries –10/+10 and a SSIP offset of 15, the difference would be larger than 10 and the synapses wouldn't move. One could say that these boundaries determine the maximum difference between events belonging to a correlated group of synapses. The 'outer' boundaries also determine the maximum firing frequencies at which the rule can be used without too many instabilities: the larger the boundaries, the lower the maximum rate. With boundaries –10/+10, firing rates up to 100 Hz perform well (average interspike interval is larger than 10 ms). For –20/+20, the 'critical' rate is 50 Hz (average interspike interval 20 ms).

The SSIP offset *o* equals the time lag between every two spikes in a sequence generated with this method. We always used an offset of 1 ms, resulting in synapses arranged one node away from each other. Is it possible to use different offsets? The answer is yes: with an offset of 4, synapses are ordered on the dendrite with three empty nodes in between. We did simulations with 3 synapses receiving SSIP input with offsets of 1, 2, 4 and 8 and synapses were always ordered correctly. (When using larger offsets, it is important to check that the LSTDP window still contains the preferred Δt and is faster than the membrane time constant.)

5.4 Correlation detection

Temporal input sequences may be reflected in hippocampal place-neurons or in the input to direction-selective neurons in the visual cortex and are therefore a realistic real world scenario, but un-patterned correlated input may reflect the stimulus-driven input in primary sensory areas. Previously, 'correlated synapses' were the synapses that fired in sequence and were therefore more correlated with the output of the neuron than independent Poisson synapses, but correlation now means that synapses fire at the same time. We will describe two types of correlation detection: 1) discrimination between correlated and uncorrelated synapses through Hat shaped LSTDP and 2) discrimination of independent groups of correlated synapses through branching.

5.4.1 Hat shaped LSTDP

The basic idea behind Hat shaped LSTDP is that synapses that are correlated with the output of the soma often achieve small spike timings: they may fire a bit late or a bit early and thus have negative or positive spike timings, but they always fire in the period in which the group and therefore the soma fire. Figure 5.3b shows that small spike timings keep synapses near the soma, while synapses with larger spike timings (either negative or positive) are moved to more distal locations. The integral of the moving away windows is larger than the integral of the moving closer window and synapses uncorrelated with the output are therefore expected to move away from the soma.

We test this with two groups of 5 synapses, of which one group receives 5 completely independent homogeneous Poisson spike trains as input and one group receives spike trains that are generated using the Multiple Interaction Method (MIP) [29]. With this model, each individual spike train is a thinned version of a single homogeneous Poisson process $w_g(t)$. Thinning is done through random deletion of spikes from this spike train, for each generated spike train

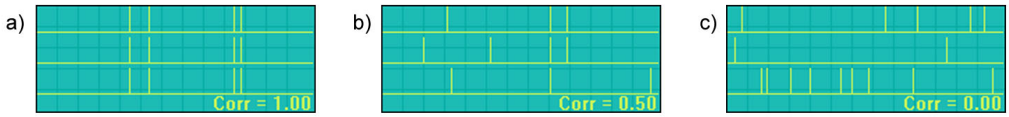

Figure 5.31. a) Three MIP generated spike trains with correlation 1.0 at 20 Hz. b) Three MIP generated spike trains with correlation 0.5 at 20 Hz. c) Three independent Poisson spike trains at 20 Hz.

individually. If the final firing rate is r and correlation is c, then the probability of deletion is $(1 - c)$ and the firing rate of $w_g(t)$ is r / c. Examples of both independent Poisson and MIP spike trains are shown in figure 5.31.

In the simulations that follow, no Weight STDP is applied, but dendritic decay is enabled: the further away from the soma synapses are, the smaller the synaptic efficacy and the longer the propagation delay between synapse and soma. This way, dendritic morphology is used to filter the inputs the neuron receives. Hat shaped LSTDP is used with default settings, the EPSP scale is set to 10, synaptic weight to 1.0 and experiments consist of 10 runs of 10.000 ms each.

Figure 5.32 shows synaptic locations during a single run: as MIP correlation in this example is 1.0, the correlated synapses always have equal spike timings and they move together, represented by the bottom line. The other five lines represent the uncorrelated synapses, clearly moving away from the soma. After 10 seconds, these synapses are approximately 40 nodes away from the soma, while the others are on the first node: these results fulfil the prediction that

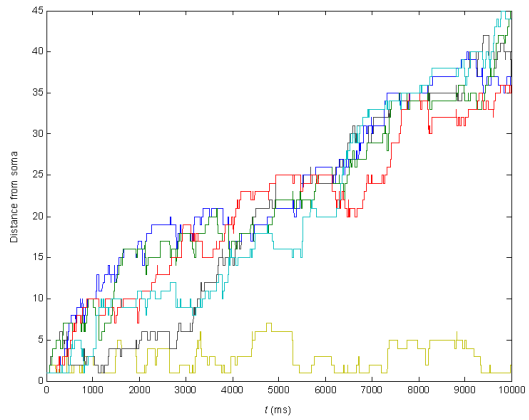

Figure 5.32. Synaptic location over time with Hat shaped LSTDP and two groups of synapses. One group keeps moving away from the soma, as it fires uncorrelated with the soma. The other group, receiving MIP input, is fully correlated and therefore moves as one (bottom line), staying close to the soma.

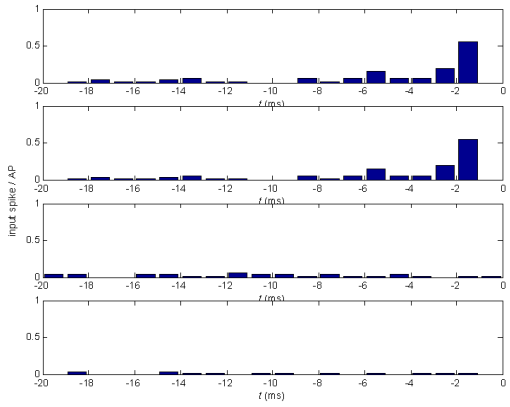

Figure 5.33. Spike triggered average input of the last 5,000 ms of a run with Hat shaped LSTDP, shown for 4 synapses. Two groups of five synapses were present, one receiving correlated inputs and one receiving uncorrelated inputs. The two synapses shown in the upper three graphs received MIP input at 20 Hz, while the bottom two synapses received independent Poisson spike trains at 20 Hz. For each synapse, the average presynaptic input just before each postsynaptic output is shown for 20 ms before each somatic AP. The 2 correlated synapses almost always fired 1 to 6 ms before an action potential was generated, while the other synapses are completely uncorrelated with the output of the neuron.

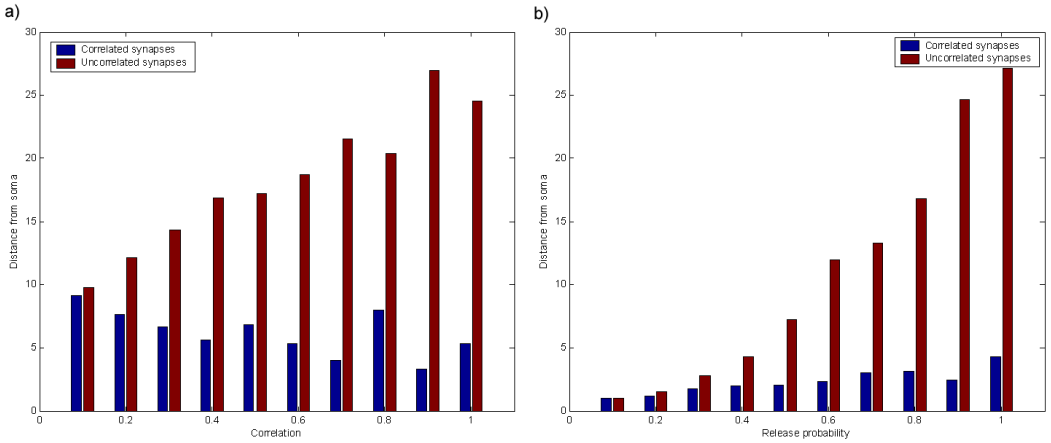

Figure 5.34. Average final distance from soma for correlated and uncorrelated synapses with different MIP correlations and release probabilities. Five correlated synapses receiving MIP input at 20 Hz, five uncorrelated synapses receiving independent Poisson input at 20 Hz. Average over 10 runs. a) Varying MIP correlation, release probability 1.0. b) Varying release probability, correlation 1.0.

correlated and uncorrelated inputs can be structurally distinguished. Note that there is no convergence: the model has no restrictions on back propagation of the APs, the uncorrelated synapses keep moving forever under this regime. The discrimination is not only visible in morphology, but also in function: there is a clear relationship between firing times of the MIP synapses and the output of the soma, which is non-existent for the uncorrelated synapses (see figure 5.33).

Varying the correlation of the MIP synapses shows that the difference in distance from the soma is a function of this correlation: as the correlation becomes larger, the MIP synapses remain closer to the soma and the other synapses are moved further away (see figure 5.34a). Decreasing the release probability affects the activity of the neuron and therefore also the distances. Unsurprisingly, higher release probability (more activity) results in larger average distances. But already with a realistic release probability of 0.6, there is a clear distance between correlated synapses and because of the dendritic decay. This has a pro-

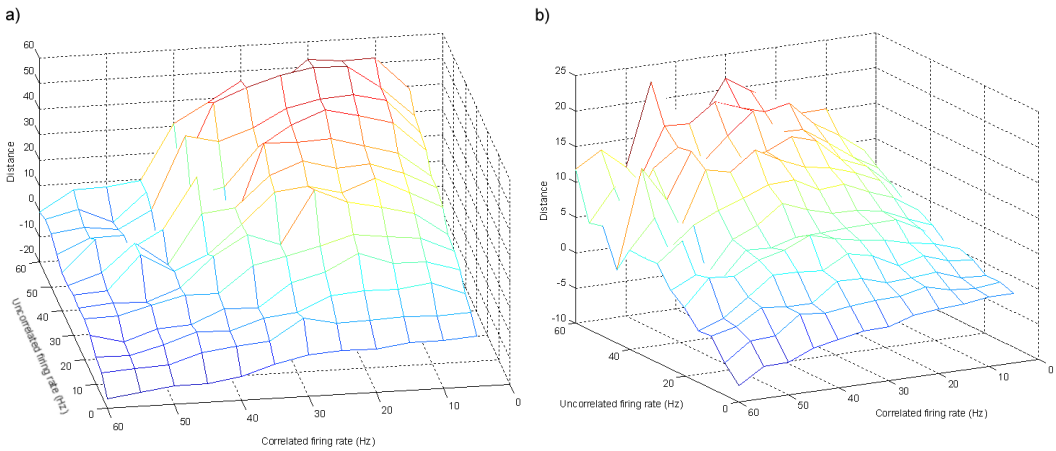

Figure 5.35. Average final distance between the two groups of synapses with varying firing rates. One group of 5 synapses receives MIP input with correlation 1.0, one group of 5 synapses receives independent Poisson inputs. Average over 10 runs. a) Release probability 1.0. b) Release probability 0.5.

found influence on synaptic efficacies (see figure 5.34b).

With all STDP rules, firing rates have an important role in determining the outcome of self-organisation: higher rates cause interactions between events and this influences behaviour. The distance between the two groups of synapses depends on the firing rates of both groups: a higher firing rate of the uncorrelated synapses causes them to move faster, which gives larger distances. Furthermore, correlated rates above 40 Hz cause the correlated synapses to move away from the soma as well and distances become smaller (see figure 5.35a; when the uncorrelated synapses fire at a very low rate, this even results in negative distances). This is with release probability 1.0: if we decrease the release probability to more realistic values, the effective firing rate of the correlated synapses is weakened and the whole 'natural' range of 5 – 50 Hz results in clear correlation detection. That is, if the uncorrelated firing rate is high enough (see figure 5.35b).

5.4.2 Branching LSTDP

Applying Weight STDP is relatively straightforward, as there are only two ways one can possibly go: depression or potentiation. Contrary, using local and unsupervised spike timing rules to relocate synapses is fairly complex: dendrites can be large trees with many branches and different signal propagation properties everywhere. Reason enough to start with a simple model: a linear dendrite over which synapses can move in two directions, herewith reducing the problem to the same dimensionality as the Weight STDP problem.

As the goal is to find rules that organise the neuron based on input and output only (and not based on chemo taxis or any other mechanism that grows a neuron unrelated to its function), we need a mechanism that makes branching useful for a model neuron. For this, we have to amend the model.

In real neurons, it has been observed that depolarisation of the dendrite attenuates the amplitude of a back propagating action potential (see chapter 2). We therefore introduce a depolarisation state s in each edge connecting two nodes in the dendrite. Whenever an EPSP is propagated forward through an edge, this state s is set to a predefined value s_{high}. Each time step, s decays exponentially with time constant τ_s. The depolarisation state now gives an indication of the time that has passed since the last EPSP passed through. We can now make back propagation of APs dependent on this state: if the state s is smaller than a threshold s_{thresh}, back propagation fails and the BPAP stops; otherwise, back propagation proceeds as normal.

It is now possible to have separate functional synapse groups that operate independently and are not bothered with BPAPs caused by other groups, in the same sense as the dendritic hot-spots proposed by Nielsen [40]. This may help to achieve and maintain optimal configurations, like two sequences of synapses that are ordered on different branches. We achieved this in simulation, but will restrict ourselves to the description of a

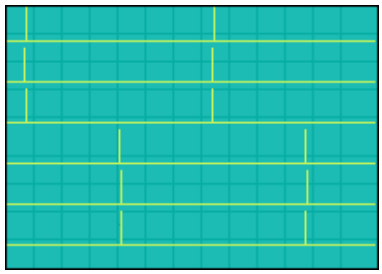

Figure 5.36. Two strongly correlated groups of three spike trains each. All synapses of each group fire within a few milliseconds, the groups fire on very regular basis and never at the same time.

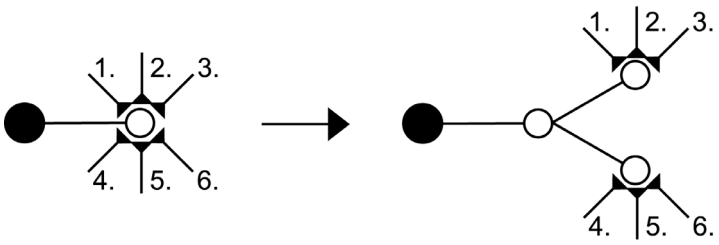

Figure 5.37. Branching example. All synapses start on the same dendrite node, but the two groups of correlated synapses are evenly divided over two dendrite branches by the Branching LSTDP rule. (See text for more details.)

thought experiment in which we separate two groups of correlated synapses.

Suppose we have a neuron with 6 synapses receiving input as shown in figure 5.36: the synapses are divided into two groups of three synapses each and all synapses in a group fire at the same time (except for some jitter), but there is always a distance between the firing of the two groups. Using the extended model and a Location STDP rule that uses only local information, it is possible to spread the synapses over two branches, as shown in figure 5.37. For simplicity, assume that we have no WSTDP, no dendritic decay and 3 EPSPs are enough to evoke an action potential.

In the initial configuration, every time one of the groups fires, the soma fires and a BPAP is initiated. This BPAP not only gets to the three synapses that caused the AP, but also to the other group. BPAPs that are caused by the correct group cause small positive spike timings (2-4 ms), but BPAPs caused by the other group causes large negative spike timings when the group itself fires next. Suppose that the distance between the two groups firing is always 50 ms, this means that both groups get spike timings of around –50 ms.

In this case, the Branching LSTDP rule counts all spike timings between –40 and –60 and when this count exceeds a certain threshold, the synapse will move away from the soma. If we suppose that synapse 1 decides to move away first, a new node has to be created and the synapse moves there. Immediately after (in the same time step or a few time steps later), both other synapses of the same group will also want to move away. How do they decide where to move? Here the edge depolarisation state s is useful: if the edge is depolarised enough (new edges start depolarised), this probably means there is a synapse there that belongs to the same group and it moves there. All three synapses of group 1 move to this node and no longer receive BPAPs caused by the other group, as back propagation fails due to a low depolarisation state. Note that BPAPs they caused themselves are still received. The other synapses are still on the initial node and receive all BPAPs. But they too will soon decide they have had enough negative spike timings and move away. But when the first one to move away checks the depolarisation of the existing edge, this is too low and thus a new branch is created. The other two move to the same branch, having a depolarisation higher than the threshold (choosing the highest value if multiple possibilities would exist). In the end, the synapse groups are on two separate branches and only get to deal with spike timings they caused themselves.

Chapter 6
Conclusions

We developed a biologically realistic model of a single neuron, inspired by current knowledge in neuroscience and artificial intelligence. The model differs from existing neuron models, as we made it abstract but didn't ignore structure: neuroscience often includes as many details as possible, which results in extremely complicated models, while artificial intelligence often models neurons as point-neurons and thus ignores any structure. We sought a compromise between these two and represented a neuron by an integrate-and-fire soma and a dendritic tree to which synapses could be attached. Plasticity is activity dependent: synaptic weight and location are controlled by spike timing dependent plasticity and dendrites elongate or branch whenever required by synaptic relocation.

We introduced a distinction between two types of STDP: Weight and Location STDP. The former controls synaptic weights, the latter is allowed to relocate synapses. Box shaped WSTDP and BPAP constrained WSTDP use the same learning window, but have different maximum weights. They can be used to establish and strengthen correlations between synaptic inputs and the output of the neuron. BPAP constrained WSTDP is better suited for achieving synaptic scaling, equalising synaptic efficacies independent of synaptic location, because the maximum weight depends implicitly on the distance from the soma.

The SSIP sequence detection task could be performed with either Box shaped LSTDP or Optimal timing LSTDP. Box shaped LSTDP uses only spike timings, but is unable to arrange the synapses in compact order. The dispersed order that can be obtained isn't very robust: unreliable synapses, synapses that are not fully correlated or adding independent Poisson synapses all have a devastating effect on performance. Nevertheless, under optimal conditions the rule keeps synapses in their preferred order in about 75% of all time steps.

Optimal timing LSTDP uses more information than Box shaped LSTDP (BPAP amplitude, a measure of distance from soma), but it orders synapses receiving SSIP spike trains in compact order. With this task BPAP constrained WSTDP is well able to equalise synaptic efficacies, scaling these appropriately to the distance from the soma. At high firing rates (> 30 Hz), performance gradually degrades, as more interactions between otherwise independent EPSPs and BPAPs occur. Performance scales almost linear with input correlation and super linear with synaptic release probability (after a certain threshold). The effect of adding independent Poisson synapses to provide background activity depends on the WSTDP regime: if we have no WSTDP and no dendritic decay (implicit synaptic scaling), adding some noise helps sequence detection, but too much noise disturbs the detection and results in bad performance. With BPAP constrained WSTDP and dendritic decay, Weight STDP depresses all uncorrelated synapses and thus effectively filters these out. This results in unaffected performance, higher uncorrelated firing rates result in even faster and better filtering. Having more synapses in a sequence (10 instead of 3) makes precise

sequence detection more difficult, as a dynamic EPSP scale is required here. As we used only static EPSP scales, noise in the form of unreliable synapses or background activity was helpful to achieve higher performances. Despite the bootstrap problems and extremely strict performance measure, the LSTDP rule was still able to achieve performances of slightly above 60% with 10 correlated synapses.

We performed un-patterned correlation detection in two different ways: discrimination between correlated and uncorrelated synapses and discrimination between two groups of correlated synapses. The first task was done with Hat shaped LSTDP and MIP generated spike trains. Because the correlated synapses have a large probability to trigger a spike in the soma, they have small spike timings on average and are kept near the soma, while the other synapses have large spike timings and keep moving away from the soma. Using this rule, the neuron can select inputs based on correlation.

For two groups of correlated synapses to become independent of each other in our model, they have to be located on two separate branches. This can be achieved with the Branching LSTDP we described, which is based on the observation that synapses that often experience large negative spike timings are subject to BPAPs they did not help trigger and should be moved to a separate branch. We described a mind experiment showing that this kind of local functional branching rule is possible, but it has quite some assumptions and it is not easy to generalise it to a more generic mechanism that works without parameter tuning.

The main question we started with is whether it is possible to achieve structural self-organisation using spike timing dependent plasticity. The Adaptive Neuron model we developed is kept as simple as possible to be able to focus on the possibilities and impossibilities of STDP, excluding effects that may be caused by other complexities in the model. Although abstract, the model is realistic and the STDP rules we used are as well: they act entirely local and unsupervised and are strongly inspired by STDP as observed in real neurons.

Using STDP in a structural model is more complicated than in point-neurons: spike timing is not only dependent on presynaptic and postsynaptic firing times, but also on the axonal and dendritic delays of EPSPs and BPAPs. In our case, synaptic distance from the soma had a large effect on spike timings. Using Weight STDP works reasonable well for small distances, but the regular STDP window (which is independent of distance) only works for certain locations. Also, synaptic scaling with strictly associative STDP can only be achieved if the maximum weight depends on the distance from the soma, which we did implicitly with BPAP amplitude in BPAP constrained WSTDP. Weight STDP is very useful as correlation filter: LTP is invoked for synapses correlated with the neuron output, LTD for uncorrelated synapses.

Because spike timings depend very much on synaptic location, it is rather difficult to functionally locate synapses on the dendritic tree using only spike timings. Sequence detection with only spike timings resulted in the dispersed order and was not robust with respect to unreliable synapses and other forms of noise. If we include the BPAP amplitude in the rule though, synapses can be positioned in the compact order and this arrangement is also pretty robust, especially in combination with BPAP constrained WSTDP. Correlation detection is easier to achieve, even with only spike timings.

Structural self-organisation using spike timing dependent plasticity is possible, but has its limits. Neural structure has a large influence on spike timings and using only these spike timings in local unsupervised learning rules poses strong restrictions on the possible tasks they can fulfil. Using more information that is locally available greatly enhances the possibilities. The two tasks we tried turned out to be feasible: both sequence and correlation detection could be accomplished within a single neuron.

Chapter 7
Discussion

We proposed a biologically realistic model of a single neuron that uses both synaptic and dendritic plasticity to achieve self-organisation. As far as we know, structural STDP is an entirely new approach and the (im)possibilities of it were completely unexplored before. Our experiments have revealed that the approach of timing rules for structural plasticity is feasible, but only to a certain extend.

Discretisation

There are some problems related to the discretisation of both time and the dendrites. With 10 synapses in a sequence, the synapse that is furthest away is already 10 nodes and therefore 10 milliseconds away from the soma. Even if LSTDP says this is the optimal location for the synapse to be, the minimal timing when an EPSP contributes to an AP is already 20 ms and falls only just in the LTP window of our WSTDP rules. Synapses can only move either one node or not at all and a single move immediately results in a change in spike timing of 2 ms, which is quite a lot. The resolution of the spike trains we use is 1 ms, so this is not really a big problem, but smaller moves are impossible and dendritic delays of more than 10 ms may be unrealistic. Also, STDP is evaluated and applied every time step, while this involves only a single network update; we feel the problem (STDP) should run on a different time scales than the neuron to be really sure that we have no effects due to quantisation. If we want the neuron to have many more synapses (100+), it may therefore be necessary to change the time step for a single network update to 0.1 ms, applying STDP only every 1 ms.

Sequence detection

The sequence detection simulations were mostly done with SSIP correlation 1.0: because this results in the best performance, it is good to be able to show the effect of other factors, but it may not be realistic that correlation is always so strong. Using only spike timings, no robust rule to order the synapses optimally could be found, because of the effect of the dendritic structure on the delays. However, with the BPAP amplitude included, synapses were ordered correctly and very robust with respect to noise, performing well under realistic conditions. More research could be done on the precise influence of the parameters on convergence speed. Also, Optimal timing LSTDP has no preference for absolute location on the dendrite and noise may cause the synapses to go rather far away from the soma; it might be worth investigating how this can be changed.

The task as we used in this thesis cannot be called sequence learning, as this traditionally means that a sequence is learned for prediction and/or generation [42,50,63]. In our model, the sequence is already present in the input

and synapses are arranged accordingly. The proposed Adaptive Neuron model is only a first step towards the design of artificial neurons with dendritic trees that self-organise based on function. It should be stressed that with the described model and learning rules, real spatio-temporal pattern recognition isn't possible yet. We looked only at structural organisation for performance, not at the output of the neuron, while the latter should be done to determine if a single neuron is able to do pattern recognition. An example of functional pattern recognition would be to have different output for two different input sequences or to notify the recognition of one single pattern by changing the output of the neuron.

Temporal sequences are very important in the brain, as they are considered to underlie many processes. Examples are direction selectivity in the visual cortex, place fields in the hippocampus and synfires. Direction selectivity is specificity of certain groups of neurons to movement on the retina in certain directions: neurons fire when movement is in the direction they are tuned to. Place fields, for example in the rat, function as an internal map, as they indicate where an organism is and can be used to find the right way in a known environment. Synfires are chains of neuron ensembles that fire in a sequence and can be important for behavioural sequence generation, like the song of a bird. We therefore think that sequence detection as we described it may be important for specific neurons.

Correlation detection

In correlation detection with Hat shaped LSTDP, the uncorrelated synapses keep moving away from the soma forever. It might be beneficial to avoid this by imposing a limit on dendrite length: otherwise, longer and larger simulations will result in extremely long dendrites and slow down the experiment unnecessarily. With a dendritic decay of 0.9, the synaptic efficacy of a synapse that is 40 nodes away is already reduced with a factor 0.9^{40} (≈ 0.015), which ought to be enough to eliminate the influence of these synapses on postsynaptic activity. A maximum dendrite length of 40+ might solve the problem of indefinite growing.

A possible application of LSTDP we haven't mentioned before is activity regulation, which is interesting in itself: with correlated synapses and Hat shaped LSTDP, synapses remain close to the soma on average, but the interactions between events in activity bursts cause them to temporally move away further, resulting in reduced synaptic efficacies, lower rates and the synapses moving back.

Using Location STDP, it is very well possible to select only those synapses that are correlated: the correlation detection we did is more functional than the sequence detection, as only correlated synapses maintained their effect on the somatic potential and this was done without Weight STDP. Although the same could be done with a hat shaped WSTDP, this wouldn't make use of the structure of the neuron, which seems biologically more realistic. The importance of discriminating between correlated and uncorrelated inputs for real neurons is quite large, as it may be very useful to distinguish real signals from noise, e.g. stimulus inputs from spontaneous background activity.

Branching

Although the Branching LSTDP rule does what it is asked, it should be considered only as a starting point for activity dependent branching. If the parameters involved are set correctly, branching and functional separation of the two correlated groups goes well, but it doesn't work without 'tuning'. Another problem is that only a mechanism for branching away is provided: moving back towards the soma is not incorporated in the rule. Re-configuration is thus not possible and branching is very liable to noise in the initial phase.

Nielsen [40] proposed 'hot-spots' as possible solution for the stability-plasticity dilemma and introduced subsets of synapses that are active one at a time. Although this made learning sequences easier, the subsets were predefined and disregarded the specific application. With our branching rule, hot-spots are created dynamically and based on the input the neuron receives. This makes the hot-spots very functional and we believe creating these hot-spots on-the-fly through spike timing dependent structural plasticity should be further investigated.

Not only hot-spots can be created dynamically this way, but complete neural structures can be 'grown' based on the input and output of the neuron. This is a large advantage compared to previous work done with branching and elongation, as this was never functional [44,46]. Adapting the neural structure by means of STDP rules with a specific input/output regime may learn us more about the functions of certain morphologies.

Consequences for machine learning

As we are dealing with only a single neuron and have only excitatory synapses, it is not easy to do tasks that are common benchmarks in machine learning (like the XOR problem). There are no local interactions within the dendritic tree yet, only the delays are affected by the structure and all inputs are simply summed. Additional modifications are required to make the neuron learn something functionally, for example to discriminate between different patterns. Information theory [7,31] may be helpful to analyse the neuron, as a single spike train (binary channel) is the only output of the neuron and any information has to be coded in this.

Despite these difficulties in making the model suitable for machine learning, we think there definitely is a future for it. As soon sequence recognition is possible, many intelligent system applications would become possible. This is especially attractive, because the model is compact, acts locally, works with discrete signals and in discrete time and is therefore relatively easy to implement in both software and hardware. And in a world where information and its flow becomes more important every day, a self-organising, unsupervised and entirely local tree-model that deals with discrete information packets might eventually be used as intelligent algorithm that automatically relocates resources to optimise communication efficiency. On the Internet, for example.

Consequences for neuroscience

In the current stage, signal propagation in the model is simplistic, but it already shows some behaviour that is interesting for neuroscience. The most important observation is probably that neurons in which location-independent STDP rules control synapses do not perform well a temporal-sequence detection task, because spike timings very much depend on synaptic location on the

dendritic tree. The commonly reported Weight STDP window with LTD with timings between –50 and –10 ms and LTP with timings between +10 and +20 ms, for example, only does what we expect from it when synapses are located 5 to 10 ms away from the soma (assuming forward and backward propagation have the same speed). We therefore expect this window to quantitatively change with the distance from the soma, like Rao and Sejnowski [49] predicted. This could be true, as spike timing measurements are always done in the pre- and postsynaptic somata, *not* in the synapse wherein STDP operates. This clearly influences the measured spike times. Furthermore, no research has been reported in which the relative locations of somata and synapse were taken into account and it certainly hasn't been investigated systematically. If the commonly reported window were location independent, STDP would never potentiate synapses that are either too close to or too far away from the soma. Another point is that using only spike timings strongly reduces the possibilities for location and weight STDP rules. However, STDP rules can only use information that is locally available in the synapse: what information can be (and is) used by synapses for plasticity? Is using the amplitude of the BPAP realistic or is there something else?

Future model improvements

There are many features that could be added to the model or could be further extended. The current dendrites, for example, are passive dendrites of the simplest kind: they linearly sum input and propagate this with a certain decay. This could be modified in many different ways to make the dendrites computationally stronger: active dendrites with signal amplification and dendritic spiking could be implemented, local interactions between EPSPs (and possibly BPAPs) in the dendritic tree could be modelled, and so on. It has been shown by many researchers that real dendrites perform complex and non-linear computations (see chapter 2) and this could be used as inspiration. We have used stochastic synapses, but dynamic synapses [34], using short term plasticity, could be used to make the system even more non-linear. Inhibitory synapses would make the neuron fairly complex, but shunting and phenomena like that could be achieved with them. In this thesis, selection of inputs is done by Weight STDP (sequence detection) or Location STDP (correlation detection), but one could also think of timing rules that do synapse splitting and retraction, herewith selecting those inputs that are useful for the neuron to obtain a certain function.

In general, the model could be modified to allow multiple neurons to be coupled to each other, allowing for much more complicated computations and dynamics. As everything is discrete, it is fairly easy to implement the model in hardware, making it computationally much more attractive to couple multiple neurons. This would also make it possible to couple a model neuron with a real neuron, giving the opportunity to see what the difference in adaptivity is between natural spike trains and the spike trains we use.

A look into the future

At the time of writing, it is completely unknown whether Location STDP as we implemented it is based on reality or not. It has been shown in the past that neural activity definitely affects structural development and some types of neurons may even keep their activity-dependent structural plasticity during their

whole life times. It is therefore certainly plausible, but spike timing dependent structural plasticity hasn't been investigated yet. We think our simulations show that there is reason enough to assume it is possible and should at least be looked into.

As we've seen before, different types of neurons come with very distinguishable dendritic trees and the functions of these morphologies are largely unknown. With our model, it may be possible to adapt neurons to a specific task and by running simulations with resulting morphologies in a more realistic model, we may in the end be able to discover more about the relationships between dendritic morphology, synaptic plasticity and input pre-processing. The brain is fascinating.

Appendix A

Adaptive Neuron parameters

The settings given below are the default settings for the model described in section 5.1. Unless stated otherwise, simulations were done with these settings.

Adaptive Neuron

Initial # nodes	1 node
Network update	1 ms

Soma (leaky integrate-and-fire)

V_{rest}	-70 mV
θ	-30 mV
τ	20 ms

Dendrites

λ	0.9
κ	0.9

Synapses

w_{init}	0.5
e	40
$p_{release}$	1.0

Location STDP

Box shaped LSTDP

Move away	[-50,+10] ms
Move back	[+20,+90] ms

Optimal timing LSTDP

Move away	[+1,+10] ms
Move back	[-10,-1] ms

Hat shaped LSTDP

Move away	[-70,-30] ms [+30,+70] ms
Move back	[-30,+30] ms

Weight STDP

Box shaped WSTDP

LTD	[-50,-10] ms -0.03
LTP	[+2,+20] ms +0.05

BPAP constrained WSTDP

LTD	[-50,-10] ms -0.03
LTP	[+2,+20] ms +0.05

Experiments

# runs	10
Run length	10.000 ms

SSIP

c	1.0
o	1 ms

MIP

c	1.0

References

1. Abbot, L.F., & Nelson, S.B. 'Synaptic plasticity: taming the beast'. In: *Nature neuroscience* 3, pp.1178-1183 (2000).
2. Becker, S. & Plumbley, M. 'Unsupervised neural network learning procedures for feature extraction and classification'. In: *Journal of Applied Intelligence* 6, pp.1-21 (1996).
3. Bell, C.C., Caputi, A., Grant, K. & Serrier, J. 'Storage of a sensory pattern by anti-Hebbian synaptic plasticity in an electric fish'. In: *Proceedings of the National Academy of Science USA* 90, pp.4650-4654 (1993).
4. Bell, C.C., Han, V.Z., Sugawara, Y. & Grant, K. 'Synaptic plasticity in a cerebellum-like structure depends on temporal order'. In: *Nature* 387, pp.278-281 (1997).
5. Bi, G.Q. & Poo, M.M. 'Synaptic Modifications in Cultured Hippocampal Neurons: Dependence on Spike Timing, Synaptic Strength, and Postsynaptic Cell Type'. In: *Journal of Neuroscience* 18(24), pp.10464–10472 (1998).
6. Bi, G.Q. & Wang, H.X. 'Temporal asymmetry in spike timing-dependent synaptic plasticity'. In: *Physiology & Behavior* 77, pp.551–555 (2002).
7. Borst, A. & Theunissen, F.E. 'Information theory and neural coding'. In: *Nature neuroscience* 2 (11), pp.947-957 (1999).
8. Chance, F.S., Abbott, L.F. & Reyes, A.D. 'Gain Modulation from Background Synaptic Input'. In: *Neuron* 35, pp.773–782 (2002).
9. Dan, Y. & Poo, M.-m. 'Hebbian depression of isolated neuromuscular synapse in vitro'. In: *Science* 256, pp.1570-1573 (1998).
10. Dayan, P. & Abbott, L.F. *Theoretical Neuroscience*. MIT Press, Cambridge, MA (2001).
11. Egger, V., Feldmeyer, D. & Sakmann, B. 'Coincidence detection and changes of synaptic efficacy in spiny stellate neurons in rat barrel cortex'. In: *Nature neuroscience* 2(12), pp.1098-1105 (1999).
12. Elman, J.L. 'Finding Structure in Time'. In: *Cognitive Science*, vol. 14, pp.179-211 (1990).
13. Engert, F. & Bonhoeffer, T. 'Dendritic spine changes associated with hippocampal long-term synaptic plasticity'. In: *Nature* 399, pp.66-70 (1999).
14. Feldman, D.E. 'Timing-Based LTP and LTD at Vertical Inputs to Layer II/III Pyramidal Cells in Rat Barrel Cortex'. In: *Neuron* 27, pp.45-56 (2000).
15. Fritzke, B. 'A growing neural gas network learns topologies'. In: Tesauro, G., Touretzky, D. S. & Leen,, T. K. (eds), *Advances in Neural Information Processing Systems* 7, pp.625--632. MIT Press, Cambridge MA (1995).
16. Froemke, R.C. & Dan, Y. 'Spike-timing-dependent synaptic modification induced by natural spike trains'. In: *Nature* 416, pp.433-438 (2002).
17. Gerstner, W. 'Spiking Neurons'. In: Maass, W. & Bishop, C. M. (eds.), *Pulsed Neural Networks*, MIT-press (1999).
18. Gerstner, W. & Kistler, W. *Spiking Neuron Models*, Cambridge University Press (2002).
19. Hebb, D.O. *The organization of behavior*. Wiley, New York (1949).
20. Hely, T.A., Graham, B. & Van Ooyen, A. 'A Computational Model of Dendrite Elongation and Branching Based on MAP2 Phosphorylation'. In: *Journal for theoretical Biology* 210, pp.375-384 (2001).
21. Hines, M.L. & Carnevale, N.T. 'NEURON: a tool for neuroscientists'. In: *The Neuroscientist* 7, pp.123-135 (2001),
22. Hopfield, J.J. & Brody, C.D. 'Learning rules and network repair in spike-timing-based computation networks'. In: *PNAS* 101, pp.337-342 (2004).
23. Horne, B.G. & Giles, C.L. 'An experimental Comparison of Recurrent Neural Networks'. In: Tesauro, G., Touretzky, D. And Leen, T. (eds), *Neural Information Processing Systems 7*, MIT Press, pp.697 (1995).

24. Johnston, D., Hoffman, D.A., Magee, J.C., Poolos, N.P., Watanabe, S., Colbert, C.M. & Migliore, M. 'Dendritic potassium channels in hippocampal pyramidal neurons'. In: *Journal of Physiology* 525.1, pp.75-81 (2000).
25. Kandel, E.R., Schwartz, J.H. & Jessell, T.M. *Principles of neural science (3rd ed)*, Prentice Hall, London (1991).
26. Kempter, R., Gerstner, W. & Van Hemmen, J.L. 'Hebbian learning and spiking neurons'. In: *Physical Review E* 59 (4), pp.4498-4514 (1999).
27. Kempter, R., Gerstner, W. & Van Hemmen, J.L. 'Intrinsic Stabilization of Output Rates by Spike-Based Hebbian Learning'. In: *Neural Computation* 13, pp.2709-2741 (2001).
28. Koch, C. & Segev, I. 'The role of single neurons in information processing'. In: *Nature neuroscience* 3, pp.1171-1177 (2000).
29. Kuhn, A., Aertsen, A. & Rotter, S. 'Higher-Order Statistics of Input Ensembles and the Response of Simple Model Neurons'. In: *Neural Computation* 15, pp.67–101 (2003).
30. Van Leeuwen, M. *Modelling timing dependent neural plasticity*. Institute for Information and Computing Sciences, Utrecht University (2003).
31. London, M., Schreibman, A., Häusser, M., Larkum, M.E. & Segev, I. 'The information efficacy of a synapse'. In: *Nature neuroscience* 5 (4), pp.332-340 (2002).
32. Maass, W., Schnitger, G. & Sontag, E. 'On the computational power of sigmoid versus boolean threshold circuits'. In: *Proc. of the 32nd Annual IEEE Symposium on Foundations of Computer Science*, pp.767-776 (1991).
33. Maass, W. & Bishop, C. *Pulsed Neural Networks*. MIT Press, Cambridge, MA (1998).
34. Maass, W., Zador, A.M. 'Dynamic Stochastic Synapses as Computational Units'. In: *Neural Computation* 11, pp.903-917 (1999).
35. Magee, J.C. & Cook, E.P. 'Somatic EPSP amplitude is independent of synapse location in hippocampal pyramidal neurons'. In: *Nature neuroscience* 3, pp.895-903 (2000).
36. Magee, J.C. & Johnston, D. 'A Synaptically Controlled, Associative Signal for Hebbian Plasticity in Hippocampal Neurons'. In: *Science* 275, pp.209-213 (1997).
37. Markram, H., Lubke, J., Frotscher, M. & Sakmann, B. 'Regulation of Synaptic Efficacy by Coincidence of Postsynaptic APs and EPSPs'. In: *Science* 275, pp.215-217 (1997).
38. Natschläger, T. & Ruf, B. 'Spatial and temporal pattern analysis via spiking neurons'. In: *Network: Computational Neural Systems* 9, pp.319–332 (1998).
39. Neville, K.R. & Lytton, W.W. 'Potentiation of Ca^{2+} influx through NMDA channels by action potentials: a computer model'. In: *NeuroReport* 10, pp.3711-3716 (1999).
40. Nielsen, B.G. 'Sequence learning in differentially activated dendrites'. In: *Network: Computation in Neural Systems* 14, pp.189-209 (2003).
41. Nishiyama, M., Hong, K., Mikoshiba, K., Poo, M.-m. & Kato, K. 'Calcium stores regulate the polarity and input specificity of synaptic modifications'. In: *Nature* 408, pp.684-687 (2000).
42. Nowotny, T., Rabinovich, M.I. & Abarbanel, H.D.I. 'Spatial representation of temporal information through spike timing dependent plasticity'. In: *Physical Review E* 68 (2003).
43. Van Ooyen, A., Pakdaman, K., Houweling, A. R., Van Pelt, J. & Vibert, J.-F. 'Network connectivity changes through activity-dependent neurite outgrowth'. In: *Neural Processing Letters* 3, pp.123-130 (1996).
44. Van Ooyen, A., Duinhouwer, J., Remme, M.W.H. & Van Pelt, J. 'The effect of dendritic topology on firing patterns in model neurons'. In: *Network: Computation in Neural Systems* 13, pp.311-325 (2002).
45. Van Pelt, J. & Uylings, H.B.M. 'Branching rates and growth functions in the outgrowth of dendritic branching patterns'. In: *Network: Computational Neural Systems* 13, pp.261–281 (2002).
46. Van Pelt, J. & Uylings, H.B.M. 'Growth Functions in Dendritic Outgrowth'. In: *Brain and Mind* 4, pp.51–65 (2003).
47. Poirazi, P. & Mel, B.W. 'Impact of Active Dendrites and Structural Plasticity on the Memory Capacity of Neural Tissue'. In: *Neuron* 29, pp.779–796 (2001).
48. Ramakers, G.J.A., Avci, B., van Hulten, P., van Ooyen, A., van Pelt, J., Pool, C.W. & Lequin, M.B. 'The role of calcium signaling in early axonal and dendritic morphogenesis of rat cere-

bral cortex neurons under non-stimulated growth conditions'. In: *Developmental Brain Research* 126, pp.163-172 (2001).
49. Rao, R.P.N. & Sejnowski, T.J. 'Spike-Timing-Dependent Hebbian Plasticity as Temporal Difference Learning'. In: *Neural Computation* 13, pp.2221–2237 (2001).
50. Roberts, P.D. 'Computational Consequences of Temporally Asymmetric Learning Rules: I. Differential Hebbian Learning'. In: *Journal of Computational Neuroscience* 7, pp.235–246 (1999).
51. Roberts, P.D. & Bell, C.C. 'Spike timing dependent synaptic plasticity in biological systems'. In: *Biological Cybernetics* 87, pp.392–403 (2002).
52. Rubin, J.E. 'Steady states in an iterative model for multiplicative spike-timing-dependent plasticity'. In: *Network: Comput. Neural Syst.* 12, pp.131–140 (2001).
53. Rubin, J.E., Lee, D.D. & Sompolinsky, H. 'Equilibrium Properties of Temporally Asymmetric Hebbian Plasticity'. In: *Physical Review Letters* 86(2), pp.364-367 (2001).
54. Rumelhart, D.E., Hinton, G.E. & Williams, R.J. 'Learning representations by back-propagating errors'. In: *Nature* 323 (1986).
55. Rumsey, C.C. & Abbott, L.F. 'Equalization of Synaptic Efficacy by Activity- and Timing-Dependent Synaptic Plasticity'. In: *Journal of Neurophysiology* 91, pp. 2273-2280 (2004).
56. Schaefer, A.T., Larkum, M.E., Sakmann, B. & Roth, A. 'Coincidence Detection in Pyramidal Neurons Is Tuned by Their Dendritic Branching Pattern'. In: *Journal of Neurophysiology* 89, pp.3143-3154 (2003).
57. Segev, I. & Rall, W. 'Excitable dendrites and spines: earlier theoretical insights elucidate recent direct observations'. In: *Trends in Neuroscience* 21 (11), pp.453-460 (1998).
58. Segev, I. & London, M. 'Untangling Dendrites with Quantitative Models'. In: *Science* 290, pp.744-750 (2000).
59. Segev, I. & London, M. 'Synaptic scaling in vitro and in vivo'. In: *Nature neuroscience* 4 (9), pp.853-855 (2001).
60. Segev, R. *Self-Wiring of Neural Networks*. M.Sc. thesis, Faculty of Exact Sciences, Tel Aviv University (1998).
61. Senn, W. 'Beyond spike timing: the role of nonlinear plasticity and unreliable synapses'. In: *Biological Cybernetics* 87, pp.344–355 (2002).
62. Sjöstrom, P.J., Turrigiano, G.G., Nelson, S.B. 'Rate, Timing, and Cooperativity Jointly Determine Cortical Synaptic Plasticity'. In: *Neuron* 32, pp.1149–1164 (2001).
63. Song, S., Miller, K.D. & Abbott, L.F. 'Competitive Hebbian learning through spike-timing-dependent synaptic plasticity'. In: *Nature neuroscience* 3(9), pp.919-926 (2000).
64. Sterrat, D.C. & Van Ooyen, A. 'Does Morphology Influence Temporal Plasticity?'. In: Dorronsoro, J.R. (Ed.), *ICANN 2002,* LNCS 2415, pp.186-191 (2002).
65. Stuart, G.J. & Häusser, M. 'Dendritic coincidence detection of EPSPs and action potentials'. In: *Nature Neuroscience* 4 (1), pp.63-71 (2001).
66. Vetter, P., Roth, A. & Häusser, M. 'Propagation of Action Potentials in Dendrites Depends on Dendritic Morphology'. In: *Journal of Neurophysiology* 85, pp.926-937 (2001).
67. Vreeken, J. *Spiking neural networks, an introduction*. Technical report UU-CS-2003-008, Institute for Information and Computing Sciences, Utrecht University (2003).
68. Wang, X.-J. 'Fast burst firing and short-term synaptic plasticity: a model of neocortical chattering neurons'. In: *Neuroscience* 89 (2), pp.347-362 (1999).
69. Williams, R.J. & Peng, J. 'An Efficient Gradient-Based Algorithm for online Training of Recurrent Neural Network Trajectories'. In: *Neural Computation* 2, pp.490-501 (1990).
70. Wong, R.O.L. & Ghosh, A. 'Activity-dependent regulation of dendritic growth and patterning'. In: *Nature reviews neuroscience* 3, pp.803-812 (2002).
71. Yeung, L.C., Blais, B.S., Cooper, L.N. & Shouval, H.Z. 'Calcium as the associative signal for a model of Hebbian plasticity: application to multi-input environments'. In: *Neurocomputing* 52-54, pp.437-440 (2003).
72. Zhang, L.I., Tao, H.W., Holt, C.E., Harris, W.A. & Poo, M.M. 'A critical window for cooperation and competition among developing retinotectal synapses'. In: *Nature* 395, pp.37-44 (1998).
73. Synapse illustration; http://www.ship.edu/~cgboeree/synapse.gif