Universität Bremen

Neuronal Models of Motor Sequence Learning in the Songbird

Maren Westkott

March 2016

Neuronal Models of Motor Sequence Learning in the Songbird

Vom Fachbereich für Physik und Elektrotechnik

der Universität Bremen

zur Erlangung des akademischen Grades eines

Doktor der Naturwissenschaften (Dr. rer. nat.)

vorgelegte Dissertation

von

Dipl. Phys. Maren Westkott

aus Wuppertal

1. Gutachter: Prof. Dr. rer. nat. Klaus Pawelzik

2. Gutachter: Prof. Dr. rer. nat. Stefan Bornholdt

Eingereicht am: 15.3.2016

Datum des Kolloquiums: 4.5.2016

Abstract

Communication of complex content is an important ability in our everyday life. For communication to be possible, several requirements need to be met: The individual communicated to has to learn to associate a certain meaning with a given sound. In the brain, this sound is represented as a spatio-temporal pattern of spikes, which will thus have to be associated with a different spike pattern representing its meaning. In this thesis, models for associative learning in spiking neurons are introduced in chapters 6 and 7. There, a new biologically plausible learning mechanism is proposed, where a property of the neuronal dynamics - the hyperpolarization of a neuron after each spike it produces - is coupled with a homeostatic plasticity mechanism, which acts to balance inputs into the neuron. In chapter 6, the mechanism used is a version of spike timing dependent plasticity (STDP), a property that was experimentally observed: The direction and amplitude of synaptic change depends on the precise timing of pre- and postsynaptic spiking activity. This mechanism is applied to associative learning of output spikes in response to purely spatial spiking patterns. In chapter 7, a new learning rule is introduced, which is derived from the objective of a balanced membrane potential. This learning rule is shown to be equivalent to a version of STDP and applied to associative learning of precisely timed output spikes in response to spatio-temporal input patterns.

The individual communicating has to learn to reproduce certain sounds (which can be associated with a given meaning). To that end, a memory of the sound sequence has to be formed. Since sound sequences are represented as sequences of activation patterns in the brain, learning of a given sequence of spike patterns is an interesting problem for theoretical considerations Here, it is shown that the biologically plausible learning mechanism introduced for associative learning enables recurrently coupled networks of spiking neurons to learn to reproduce given sequences of spikes. These results are presented in chapter 9.

Finally, the communicator has to translate the sensory memory into motor actions that serve to reproduce the target sound. This process is investigated in the framework of inverse model learning, where the learner learns to invert the action-perception cycle by mapping perceptions back onto the actions that caused them. Two different setups for inverse model learning are investigated: In chapter 5, a simple setup for inverse model learning is coupled with the learning algorithm used for Perceptron learning in chapter 6 and it is shown that models of the sound generation and perception process, which are non-linear and non-local in time, can be inverted, if the width of the distribution of time delays of self-generated inputs caused by an individual motor spike is not too large. This limitation is mitigated by the model introduced in chapter 8. Both these models have experimentally testable consequences, namely a dip in the autocorrelation function of the spike times in the motor population of the duration of the loop delay, i.e. the time it takes for a motor activation to cause a sound and thus a sensory activation and the time that this sensory activation takes to be looped back to the motor population. Furthermore, both models predict neurons, which are active during the sound generation and during the passive playback of the sound with a time delay equivalent to the loop delay. Finally, the inverse model presented in chapter 8 additionally predicts mirror neurons without a time delay. Both types of mirror neurons have been observed in the songbird [GKGH14, PPNM08], a popular animal model for vocal imitation learning.

Contents

1 Introduction **9**

2 Biological Background **15**
2.1 Neurons . . . 15
2.1.1 Neurons . . . 15
2.1.2 Synapses . . . 17
2.1.3 Plasticity . . . 18
2.1.3.1 STDP . . . 19
2.2 Songbirds . . . 19
2.2.1 Behaviour . . . 20
2.2.2 Basic Neuroanatomy . . . 20
2.2.3 Mirror Neurons . . . 21

3 Theoretical Background **23**
3.1 Neuron Models . . . 23
3.1.1 Rate Neurons . . . 24
3.1.2 Spiking Neurons . . . 24
3.1.2.1 The Integrate-and-Fire Neuron . . . 24
3.1.2.2 The conductance-based Integrate-and-Fire Neuron . . . 26
3.1.2.3 The Hodgekin-Huxley Neuron . . . 26
3.2 Noise . . . 27
3.3 Network Models . . . 28
3.3.1 Feed-Forward Networks . . . 28
3.3.2 Recurrent Networks . . . 29
3.4 Learning . . . 29
3.4.1 Unsupervised Learning . . . 30
3.4.1.1 Hebbian Learning Rule . . . 30
3.4.1.2 Spike Pair Spike Timing Dependent Plasticity . . . 31
3.4.2 Supervised Learning . . . 32
3.4.2.1 The Perceptron . . . 32
3.4.2.1.1 Perceptron Learning Rule . . . 32
3.4.2.2 The Tempotron . . . 33
3.4.2.3 The Chronotron . . . 33
3.4.2.3.1 The δ-rule and ReSuMe. . . . 34
3.4.2.3.2 E-Learning. . . . 34
3.4.2.3.3 FP-Learning. . . . 35
3.4.3 Reinforcement Learning . . . 35

3.4.3.1 Learning in Recurrent Networks . . . 37
3.4.3.1.1 Hopfield networks . . . 37
3.4.3.1.2 Temporal sequences of patterns . . . 37
3.5 Spike Train Dist. . . . 39
3.5.1 VanRossum-Distance . . . 39
3.5.2 Victor-Purpura-Distance . . . 39

4 Theory of Songbird Learning **41**
4.1 Reinforcement Learning . . . 41
4.2 Inverse Models . . . 42
4.2.1 Existing Learning Rules . . . 43
4.2.2 Inverse Models in Spiking Neurons as a Form of Pattern Association 44

5 Inverse Models with RSTDP **47**
5.1 Introduction . . . 47
5.2 The Model . . . 48
5.2.1 Neuron Model . . . 49
5.2.2 World Model . . . 49
5.2.3 Spike-Timing Dependent Plastictiy . . . 50
5.2.4 Measuring the Learning Progress . . . 50
5.2.4.1 General Measuring Procedure . . . 50
5.2.4.2 Measure of Pattern Similarity . . . 52
5.2.4.3 Spike Train Distance Measures . . . 52
5.2.5 Autocorrelation Function . . . 53
5.3 Results . . . 53
5.3.1 Intuitive Understanding of the Learning Process . . . 53
5.3.2 Quantitative Evaluation of the Learning Process . . . 54
5.3.3 Dependency on System Size . . . 56
5.3.4 Dependency on Firing Rates . . . 57
5.3.5 Learning with Background Noise . . . 57
5.3.6 Necessity of Exploration with Testing Firing Rate . . . 58
5.3.7 Towards more complex Models of the World . . . 58
5.3.8 Experimentally testable Predictions: the Spike Autocorrelation . . . 61
5.4 Discussion . . . 61

6 Perceptron **65**
6.1 Introduction . . . 65
6.2 The Model . . . 67
6.2.1 Neuron Model and Network Structure . . . 67
6.2.2 The Plasticity Rule . . . 67
6.3 Equivalence to PLR . . . 69
6.4 Discussion . . . 70

7 Chronotron **73**
7.1 Introduction . . . 73
7.2 The Model . . . 75
7.2.1 Neuron Models . . . 75
7.2.1.1 The simple Integrate-and-Fire Neuron . . . 75

7.2.1.2 The conductance-based Integrate-and-Fire Neuron 76
7.2.1.3 The Hodgkin-Huxley-type Neuron 76
7.2.2 Learning Rule 77
7.2.2.1 Chronotron Setup 78
7.3 Results 80
7.3.1 Membrane Potential Dependent Plasticity 80
7.3.2 Homeostatic MPDP on Inhibitory Synapses is compatible with STDP 80
7.3.3 Homeostatic MPDP allows Associative Learning 81
7.3.4 Associative Learning with a real Teacher 82
7.3.5 Associative Learning in the conductance-based Integrate-and-Fire Neuron 82
7.3.6 Associative Learning in a Hodgekin-Huxley-type Neuron 84
7.3.7 Other Results on MPDP 84
7.4 Discussion 84
7.4.1 Biological Plausibility of MPDP 87
7.4.2 Properties and Capabilities of Homeostatic MPDP 88
7.4.3 Relation of MPDP to other Learning Rules 89

8 Inverse Models with MPDP 93
8.1 Introduction 93
8.2 The Model 94
8.2.1 Network Setup 94
8.2.2 Neuron Model 95
8.2.3 Learning Rule 96
8.2.4 World Model 96
8.2.5 Measuring the Learning Progress 96
8.2.5.1 General Measuring Procedure 96
8.2.5.2 Measure of Pattern Similarity 97
8.2.5.3 Spike Train Distance Measures 98
8.2.6 Autocorrelation Function 98
8.3 Results 98
8.3.1 Basic Learning Mechanism 98
8.3.2 Quantitative Evaluation of the Learning Process 100
8.3.3 Dependency on System Size for Different World Models 101
8.3.4 Dependency on Firing Rates 103
8.3.5 Necessity of Exploration with Testing Firing Rate 104
8.3.6 Experimentally testable Predictions 104
8.3.6.1 Autocorrelation Function 104
8.3.6.2 Mirror Neurons 105
8.4 Discussion 106

9 Recurrent Networks 109
9.1 Introduction 109
9.2 The Model 110
9.2.1 Network setup 110
9.2.2 Neuron Model 110
9.2.3 Learning Rule 111

9.2.4 Evaluation of Learning Success . . . 111
9.2.4.1 Measure of Pattern Similarity . . . 112
9.2.4.2 Spike Train Distance Measures . . . 112
9.2.4.3 Measure of Learning Success . . . 112
9.3 Results . . . 113
9.3.1 Intuitive Understanding of the Learning Process . . . 113
9.3.2 Quantitative Evaluation of the Learning Process . . . 115
9.3.3 Scaling with the System Size . . . 115
9.4 Discussion . . . 116

10 Discussion **119**
10.1 Summary and Discussion . . . 119
10.2 Outlook and Future Work . . . 123

Bibliography **131**

Chapter 1

Introduction

Throughout this thesis, as the writer I am communicating with the reader in the form of written communication. Written communication is based on language which evolved as a form of oral communication. By the means of speech, complex concepts can be transferred from one person to another. This enables cooperative approaches to developing further, even more complex concepts as well as technology. Hence, vocal communication is one of the fundamental properties that make us human.

To be able to verbally communicate with one another, people have to convert sound into meaning. To that end, sounds are first converted into activation signals in the brain. The functional cells of the brain are called neurons, which send signals from one neuron to an other via their connections called synapses. In each neuron, these signals are coded in the membrane potential, which is the voltage difference across the membrane. When a neuron is active, the membrane potential produces a sharp increase, followed by a sharp decrease, a process called an action potential or spike. Sounds are thus converted into elongated patterns of spikes in a number of auditory neurons. To be able to derive meaning from these spike patterns, patterns have to be assigned a certain meaning. This process is called associative learning.

In theoretical studies the problem of associative learning has been discussed in different contexts. The patterns that are the most simple to study are purely spatial, i.e. they consist of a pattern of activation or non-activation in different neurons. These neurons provide input into an output neuron, that is taught to associate one type of output with one group of the input patterns and a different type of output to another group, the groups being predefined from the outside. This problem is called the Perceptron problem. It can easily be generalized to the temporal domain: Here, a output neuron is required to classify spatio-temporal input patterns into two groups by either spiking in response to a pattern or not spiking. This problem is called the Tempotron. An even more sophisticated problem, which is probably most closely related to the real situation in biological neurons, is the associative learning of precisely timed output spikes in response to spatio-temporal patterns. This is called the Chronotron problem.

In modelling studies on neural networks, learning rules have been devised that enable a neuron to learn these association tasks. However, these learning rules suffer from varying degrees of lack of biological realism. One particular problem all these learning rules have

in common is that it is unclear, how the neuron can be instructed on what the correct response to a given input is, a signal called the teacher signal.

In addition to associating meaning with a certain sound, communication requires the reproduction of learned sounds. This process is called vocal imitation learning. Vocal learning is a property that separates humans from other primates. It is shared with very few mammals, such as whales, and with songbirds. Understanding the processes that underlie vocal imitation learning is a fascinating endeavour. To be able to learn something about the neuronal activations that occur during vocal imitation learning, experimental studies have been done on songbirds. The vocal imitation learning process of songbirds shares a remarkable amount properties with the one of humans: There is a critical phase, in which the sounds which are to be imitated are learned. In this phase, social interactions are very important [BD13]. After that, in a babbling phase, the young learner produces first, small sound segments, which over time become more and more similar to the target sound sequence. Since it is easier to access experimental data for songbirds than for humans, theoretical studies may also focus on songbird imitation learning in a first step. Due to the similarities in the learning process, however, it can be hoped that some of the general principles that are found to enable songbirds to perform imitation learning are also present in humans.

The first step to vocal learning is the ability to memorize a sequence of sounds. This sequence of sounds is represented in the brain in a sequence of activations in an auditory brain area. Memorizing the sequence is then equivalent to being able to reproduce the sequence without the auditory input. In theoretical neuroscience, there have been advances towards the learning of reproducing sequences of activations in artificial neuronal networks [Hop82]. These are usually connected in an all-to-all fashion, such that each neuron gives input to all other neurons in the network and thus indirectly to itself. In these so-called recurrent networks, sequences can be in general conceived in the following way: A neuron activates the next neuron, which in turn activates the neuron after that and so on, until the sequence is complete. However, if more than one sequence is supposed to be learned, it is important, that each neuron does not only get input from one neuron, but from several. This implies, that the combined input from several neurons will activate a subsequent neuron. Thus, if these input neurons are not activated exactly in the correct way, it is possible that they fail to activate the subsequent neuron and thus, the sequence is interrupted. Likewise, if one part of the sequence is slightly distorted, more than just the target neuron can be activated, which can lead to a very high network activity. Thus, recurrent networks are highly sensitive to noise. Therefore, it is difficult to generate, let alone learn, stable sequences of activation. It has been shown that it can be done, however, if during the learning process the network is already exposed to noise, which enables it to learn to tolerate some degree of noise and thus generate stable sequences [LB13]. However, most of the learning rules employed are relatively artificial and not suitable to imprint sequences of activation on networks of realistic spiking neurons.

Once the auditory sequence is memorized, it needs to be translated into suitable motor activations that in turn generate the same sound sequence. It has been hypothesized that this is done in the form of reinforcement learning, where the young learner just tries out variations of muscle activation and gets some feedback on how well she is performing.

From this signal, the learner can change her behaviour towards better performances by trial-and-error learning. This framework works well for simple, low-dimensional targets, but suffers in a high-dimensional setting, where there are just too many different types of motor activation that would have to be tried out. Furthermore, each new sequence would have to be learned individually from scratch. Since imitation learning is remarkably fast in humans, such that a simple tune can be imitated immediately upon hearing it only once, this can not be the only mechanism enabling humans to perform vocal imitation learning. Some songbirds also learn to imitate remarkably complex sounds, such as the lyrebird for example, which is famous for imitating car alarms and camera shutters as well as chain saws. It is hard to conceive how this could be achieved with reinforcement learning.

Another way to learn to generate a motor sequence from the memorized sound sequence is provided by so-called inverse models. There, in the learning phase the young learner produces arbitrary sounds, similar to the babbling phase of human babies and young songbirds. During this phase she experiences the forward model of "this particular motor activation produces this sound and from that derives the inverse model "if I want to produce this sound, I have to perform this particular motor activation". These inverse models have been suggested as a learning mechanism [HGH14]. It was shown that for a simple correlational learning rule and a linear forward mapping, a highly variable motor code during the learning (or exploration) phase leads to a so-called causal inverse model, which maps auditory activations back onto their respective motor causes. It remains an open question however, how more complex action-perception mappings on realistic spiking neurons could be inverted with a biologically plausible learning rule. Such a learning rule faces two main difficulties: In the motor population, which triggers the sounds and therefore the auditory activation, a trace of earlier activations has to persist over time, until the activation from the auditory population is fed back into the motor population. Furthermore, then the original activation and the feedback from the auditory population have to be compared.

Learning of inverse models or classifications or, in fact, of anything else has to manifest in a physiological change in the brain. The underlying hypothesis in neuroscience is that this physiological change is in the strength of the synapses, the connections between neurons. How much each neuron is influenced by the activation of another neuron providing input to it is different between neurons. The magnitude of this influence is experimentally accessible, so the strength of the coupling between to neurons can be measured. This connection strength is called synaptic strength, efficacy or weight. The synaptic efficacy can change over time due to activations in the two neurons forming the synapse. The hypothesis that learning is just the changing of the synaptic efficacy was formulated by Hebb [Heb49]:

> Let us assume that the persistence or repetition of a reverberatory activity (or "trace") tends to induce lasting cellular changes that add to its stability. When an axon of cell A is near enough to excite a cell B and repeatedly or persistently takes part in firing it, some growth process or metabolic change takes place in one or both cells such that A's efficiency, as one of the cells firing B, is increased.

This implies that previous activations in the brain drive synaptic change. This process is

called synaptic plasticity.

Synaptic plasticity is investigated extensively in experimental studies. It has been found that the direction of synaptical change depends on the precise timing of spiking activity in both neurons [DP04, DP06, CD08]. This phenomenon is called Spike Timing Dependent Plasticity (STDP): the neuron sending a signal via the synapse, which is also called the presynaptic neuron and the neuron receiving the signal, which is also called the postsynaptic neuron. In a typical excitatory neuron, the intensity of the weight change is stronger the closer in time the two activations are. For a causal spike pattern, where first the presynaptic neuron spikes and then the postsynaptic neuron spike, the synaptic weight is strengthened and for an anti-causal spike pattern, the weight is weakened. More recently, however, it has been found that the direction and strength of the weight change depends on the activation patterns in a much more complex way [FD02, WGNB05, SH06, AWP13]. Additionally, it is unclear what the computational purpose of these synaptic learning rules is. Theoretical work can thus provide meaningful insights into the nature and purpose of synaptic learning rules. Furthermore, it can identify new learning rules that are useful from a computational perspective and thus inspire new experimental investigations.

In theoretical studies, the role of plasticity is often investigated independently from the underlying dynamics of the neuronal network. Combining for example classic STDP with any network of spiking neurons leads to an unbounded strengthening of the synaptic weights, which are strong enough to cause spikes, since each presynaptic activation leads to a causally timed postsynaptic activation, which leads to a strengthening of the synapses, which in turn even strengthens this effect. These runaway effects can be avoided by a capping of the weights, such that weights grow to a maximal weight but not beyond. Another possibility is the introduction of a self-inhibiting state of each neuron after a spike. This limits the total number of spikes. Homeostatic plasticity mechanisms are yet another example of synaptic plasticity mechanisms that aim to keep the activity in a given network at a target activity.

Finally, it is important to take the interaction of the learning system with the outside world into account. Many modelling studies focus on one particular brain area, which then is modelled in great detail. However, complex behaviour relies on the interaction of different brain areas, which can be modelled as different modules. Looking at the interactions of the modules and how their combined activations interact with the world can provide meaningful insights into the functional modules necessary to perform a given task. From these theoretical insights, new directions for experimental investigations can be derived.

This thesis is organized in ten chapters. The first chapter provides an introduction to the topic, the second chapter gives a brief overview of the biological context, which inspires the theoretical work. There, the core pieces of information on neurons, synaptic plasticity and on vocal imitation learning in the songbird are provided. In the following chapter, the theoretical considerations that are the basis for this work, will be introduced. This includes a number of neuron and plasticity models. Furthermore, the concept of supervised learning is introduced. In chapter four, the theoretical models that have been devised to investigate songbird imitation learning will be presented. Furthermore, a underlying basic

insight into the nature of imitation learning on spiking neurons is introduced. In the fifth chapter, a simple inverse model learning algorithm based on the self-inhibition of neurons after a spike and a non-classical form of STDP is introduced. This learning mechanism is applied to a simple Perceptron classifier in chapter six. The extension of the Perceptron to the temporal domain, the Chronotron is investigated in chapter seven, where a new learning rule based on a homeostatic principle is introduced. In chapter eight, the analogy of Chronotron learning to the learning of inverse models in spiking neurons that was intoduced in chapter four is applied to learning inverse models with the previously introduced learning rule. Furthermore, this learning rule is applied to the learning of sequences in recurrent networks in chapter nine. Finally, all results are summarized and discussed in chapter ten.

To facilitate understanding of individual chapters independently from each other, the description of all relevant parts of the respective models will be repeated in each chapter.

Chapter 2

Biological Background

In this chapter, I will provide a brief overview over the biological background of this thesis. The basic function of neurons, the functional cells in the brain of animals, will be introduced. The connections between these neurons, which provide the basis for the computing power of the brain, will be discussed. Additionally, I will introduce basic concepts of the phenomenology of how the connections change over time as a response to experience.
Finally, I will briefly introduce the biological model system, on which this thesis is focused: the songbird. The behaviour of songbirds is diverse and varies greatly between species. For theoretical work, abstractions on this diversity are necessary in order to restrict the description to the properties of a generic vocal learner. Therefore, the description here will also be limited to the generic learning behaviour of songbirds. Finally, the functionally important parts of the songbird brain anatomy, the song system, will be introduced briefly. The brain function of the songbird is an active field of experimental research. Therefore, theoretical work can contribute to understanding the functionally important modules a vocal learner needs to have and to match these onto experimental results. Finally, I will focus on the phenomenon of mirror neurons in the songbird brain, which are neurons that are active in the same way when the bird sings its song and when it passively hears the same song. These mirror neurons have inspired new theoretical work on vocal learning [HGH14].

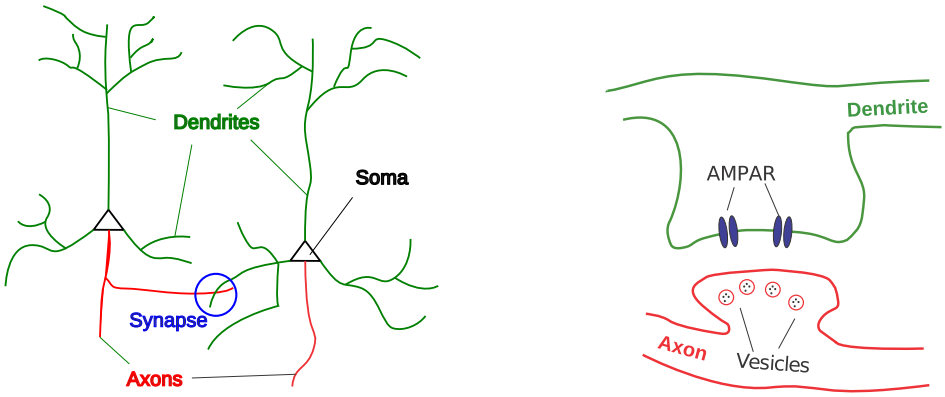
2.1 Neurons and Neuronal Networks

In this section, I will provide a brief introduction to biological neurons and their mutual connections. This description loosely follows [DA01].

2.1.1 Neurons

Neurons are the main functional cells that allow the brain to perform computing tasks. They differ from other cells in animals based on two main characteristics: Their shape and the fact that they have a voltage potential across the cell membrane.
The morphology of neurons typically consists of a cell body and two different types of appendages: Dendrites and axons (see figure 2.1(a) for a schematic drawing). When a dendrite of one neuron is close to the axon of another neuron, they can form a connection,

a so-called synapse. Via this connection, activation signals can travel from one neuron to the other. The connection is directed, that is signal travel from the axon of the "sender" neuron to the dendrite of the "receiver" neuron, but not in the other direction. Due to this direction of information flow, the "sender" neuron is also called pre-synaptic neuron and the "receiver" neuron is called post-synaptic neuron. Axons are smooth appendices that only branch out at the end, while dendrites branch out a lot to collect input from a high number of presynaptic neurons.
The inside of a neuron is separated from the outside by the cell membrane, a bi-lipid layer. Across this membrane, a potential difference can be measured, which is called the membrane potential. This membrane potential is the medium of signal processing and the basis for fast signal transmission along the axon. It is kept in place by ion pumps, which use metabolic energy to maintain a gradient of calcium, potassium and sodium ions across the membrane. Concentrations of calcium and potassium are higher on the outside of the cell, while the sodium concentration is higher on the inside. In equilibrium, the voltage across the membrane hovers around $-70mV$. This is called the resting potential. Ion channels in the membrane can disturb this equilibrium: They can open to let ions of a specific type flow along the gradient. Voltage dependent ion channels open and close depending on the membrane potential. When the membrane potential is sufficiently perturbed away from the equilibrium, a fast feedback process is triggered, such that the voltage-dependent ion channels behave in a stereotyped way to allow for the membrane potential to perform a sharp increase, followed by a sharp decrease with an undershoot of the membrane potential below the resting potential. This stereotyped response is called an action potential or, due to the short duration of the perturbation, a spike. This action potential is triggered, when the membrane potential is sufficiently high, that is above a certain spiking threshold, which is typically at about $-50mV$. At the spiking threshold, first the sodium channels open, allowing positive sodium ions to flow into the cell, thus depolarizing the neuron. Then the membrane potential quickly rises (in less than a millisecond) to about $0mV$. Slightly later, the potassium channels open, causing an influx of potassium which leads to a sharp drop in the membrane potential. This drop stops at the reset potential, which is typically below the resting potential. From here, the membrane potential relaxes back towards the resting potential. The whole process of action potential generation is very fast, such that the action potential is localized within the cell. They are usually triggered in the cell body (soma), where the input from other neurons is integrated, and then quickly travel along the axon towards the synapses and also along the dendrites. The signals travelling from the soma back up the dendrite are functionally different from the action potentials travelling along the axon and are called backpropagating action potential. After travelling down the axon upon arrival at the presynapse, the strong depolarisation of the action potential triggers synaptic transmission. Hence, the action potential is the fast travelling signal that allows for rapid information transmission between neurons.
When the strong depolarization of the action potential invades the presynaptic bouton, neurotransmitters are released into the synaptic cleft between the presynaptic bouton and the postsynaptic spine. These neurotransmitters bind to the ligands of the other class of ion channels in the membrane, which for that reason are called receptors. These are strategically positioned at the postsynaptic spine to facilitate synaptic transmission. These ion channels are specific to the type of ion that they allow to pass through as well. Hence, the influence on the membrane potential of the transmission of the signal across the synapse can either be towards more or less polarized values. When the membrane potential



(a) (b)

Figure 2.1: Simplified schematic drawing of a neuron and a synapse. Left: Two neighbouring neurons; each neuron receives input from other neurons through its dendrites, integrates these in the soma and, upon a spike sends input to other neurons via its axon. When the dendrite of one neuron comes close to the axon of another one, a synapse can form. (b) On the presynaptic side, a bouton forms, where vesicles containing neurotransmitters are stored. Upon arrival of a spike at the synapse, these vesicle fuse with the membrane to release the neurotransmitters into the synaptic cleft. There, they will bind to the receptors at the postsynaptic spine, which will in turn open the ion channels, thus causing a deflection of the membrane potential.

is deflected upwards towards the spiking threshold, the input into the neuron is called excitatory. An inhibitory input into a neuron is caused by an ion channel, which allows for the membrane potential to drop in response to the binding of the neurotransmitter, thus pushing the neuron away from the spiking threshold, or inhibiting it from spiking. The disturbance of the postsynaptic membrane potential is called postsynaptic potential or PSP (either excitatory or inhibitory). The maximum amplitude of the PSP varies between synapses, which implies that synapses are of different strength.
When a signal is transmitted through the synapse into the dendrite of the postsynaptic neuron, it travels towards the soma, where all inputs are integrated. When the summed input currents suffice, the neuron generates its own spike which is transmitted to its postsynaptic neurons.

2.1.2 Synapses

Whenever the axon of one neuron comes close to the dendrite of another one, a synapse may form (see figure 2.1(b) for a schematic drawing). There are two types of synapses: Chemical synapses and electrical synapses. While in electric synapses, a bidirectional and very direct connection is formed, in chemical synapses a more intricate form of signal transmission is employed. To that end, a physical structure at the site of the connection forms: On the presynaptic side a bouton forms, which gets its name from the button-like shape. On the dendrite, a spine forms, which is elongated towards the axon. Between the

presynaptic bouton and the postsynaptic spine, there is the synaptic cleft, which separates the two by about $20 - 50nm$. In the presynaptic bouton, there are vesicles, little bubbles of cell membrane filled with neurotransmitters. When a presynaptic spike arrives at the presynapse, these vesicles can fuse with the cell membrane, releasing neurotransmitters into the synaptic cleft. If they do, the neurotransmitters can diffuse very quickly towards the postsynapse, where they bind to receptors, which in turn open their ion channels. This elicits a perturbation in the membrane potential, which then travels towards the soma of the postsynaptic neuron. Neurotransmitters unbind from the receptors and are removed from the synaptic cleft, as they are reuptaken into the presynapse.
While the transmission of the main electrical signal is unidirectional, there are retrograde messengers that travel backwards from the postsynapse to the presynapse. These messengers are not involved in computational tasks, but they are only important for plasticity purposes.
There are two basic types of synapses: Those that cause a upwards deflection of the postsynaptic membrane potential upon a presynaptic spike are called excitatory. Those that cause a downwards deflection are called inhibitory. The former is implemented by sodium channels, while the latter is implemented by potassium channels. The receptors of these different ion channels react to different neurotransmitters, such that the neurotransmitter the presynapse emits has to match the postsynapse. The majority of excitatory synapses is formed by glutamatergic synapses, which are activated by the neurotransmitter glutamate. The matching receptors are called AMPA receptors, which open sodium channels. Inhibitory synapses are usually GABAergic synapses, which are governed by the neurotransmitter GABA (γ-aminobutyric acid). The ion channels which are opened upon GABA exposition are potassium channels. Which synapse can form between two neurons is defined by the presynaptic neuron: The output from a neuron is either excitatory or inhibitory. This implies, that synapses cannot change their type and therefore the direction of influence on the membrane potential. This property is called "Dale's law". However, due to the very large number of synapses in the brain, each neuron receives both excitatory and inhibitory inputs at all times.
Synapses also differ from each other in strength. To evaluate the strength of a synapse experimentally, an experimentator first needs to find a pair of connected neurons. Then, a spike in the presynaptic neuron is triggered, which is transmitted to the postsynaptic neuron via the synapse in question. In the postsynaptic neuron, the deflection of the membrane potential in response to the single presynaptic spike can be measured. The amplitude of this postsynaptic potential then depends on the strength of the synapse. This synaptic strength, or weight, provides an estimate of how much the activity of the presynaptic neuron influences the postsynaptic neuron. The synaptic weight is not stationary over time, but can change due to pre- and postsynaptic activity. This process of changing synapses is called synaptic plasticity and will be discussed in the next section.

2.1.3 Plasticity

The strength of a synapse is defined by the strength of the deflection of the postsynaptic membrane potential upon a presynaptic activation. This connection strength is subject to changes over time. In fact, it is assumed that the changes in strength of synapses is what enables learning in the brain. The strength of a synapse can change on different time scales: Upon repeated presynaptic activation, *short term plasticity* takes place, usually in

form of a weakening of the synaptic transmission. This can be seen in the decrease of the amplitude of the PSP in response to a high frequency presynaptic burst. This is caused by a depletion of neurotransmitter vesicles in the presynapse. It takes a few seconds to refill the vesicles and return the synapse to the original strength. In this thesis, effects of short term plasticity will be omitted.
The other type of synaptic plasticity is called *long-term plasticity*, because it leads to lasting changes in the strength of the synapse. This is the kind of synaptic change that is hypothesized to be involved in learning. The long-term weakening of a synapse is called long-term depression (LTD), while the long-term strengthening of a synapse is called long-term potentiation (LTP). LTP and LTD can be a result of very short pre- and postsynaptic activity and have long lasting effects. The dependency of plasticity on pre- and postsynaptic firing patterns is diverse and complex. One striking characteristic is the dependency of the synaptic change in response to pre- and postsynaptic changes on the precise temporal order of these spikes, a phenomenon that is called spike-timing dependent plasticity and will be discussed in the next section.

2.1.3.1 Spike Timing Dependent Plasticity

In modern biological experiments, it was found that the nature of synaptic change depends on the specific activation pattern of pre- and postsynaptic changes. This phenomenon, spike timing dependent plasticity (STDP), was first investigated in spike pairs [DP04, DP06, CD08], where it was found that for excitatory synapses the causal spiking order (first pre- and then postsynaptic) leads to a synaptic strengthening, while the reverse order leads to a weakening of the synapse. The overall amplitude of the synaptic change depends on the temporal distance between pre- and postsynaptic change and decays approximately exponentially with that distance; for a simple computational model see section 3.4.1.2. Since this is the Classical shape of STDP, this type of STDP will be called CSTDP throughout this thesis. However, experiments with more complex spike patterns have revealed that STDP is much more complex than that. For example, it was found that for inhibitory synapses, the causal order of spikes induces a strengthening, while the anti-causal order induces a weakening, which has the opposite net effect of the standard shape of excitatory STDP on the membrane potential [HNA06]. Since this net effect is the exact opposite of the classical shape of CSTDP, I will here call it reverse STDP (RSTDP). In excitatory synapses, it was found that the reversed temporal order (first post- then presynaptic spiking) could lead to LTP (and vice versa; RSTDP), depending on the location of the synapse on the dendrite [FPD05, SH06]. Additionally it has been shown that CSTDP does not always rely on spikes, but that strong subthreshold depolarization can replace the postsynaptic spike for LTD, while keeping the usual timing dependence [STN04].
Due to its compelling temporal dependencies and relative simplicity of modelling, STDP has been a popular learning rule investigated in theoretical studies.

2.2 Songbirds

Imitation learning is central to learning from others in a group. While imitation learning in general is hard to investigate, because the behaviour is hard to quantify, vocal learning is a comparatively well controlled situation. Songbirds are one of the very few animals that

perform vocal learning, which makes them a good animal model for studying the details of how vocal learning comes about. In fact, there are remarkable similarities between vocal learning in songbirds and humans [DK99, Mar70, Moo09]. A recent review discussing the possibilities of the bird model in applied medical research is given in [BD13].

In this section, I will provide a brief introduction into the learning behaviour in songbirds. This process involves several sets of different learning phases, in which the bird needs to be exposed to the right kind of stimuli for the learning process to be effective. I will go on to discuss the neuroanatomical basis of imitation learning in songbirds, the avian song system. While a lot of studies have been done to investigate the connections between brain areas and their specific roles, the precise structure and the specific task of each brain area remain under investigation. Here, theoretical work can provide insights into possible roles of different brain areas that are necessary for vocal imitation learning. Lastly, I will focus on one specific phenomenon in the avian brain: mirror neurons. These are neurons that are active in the same way when the bird vocalizes and when it passively listens to a playback of its song [PPNM08, GKGH14]. These mirror neurons give rise to theoretical considerations linking songbirds learning to inverse models (see section 4) [HGH14]. Good reviews on the vocal learning in songbirds are given in [BD02] and [DK99].

2.2.1 Behaviour

Songbirds use their songs to woo a mate and defend a territory. Young songbirds therefore learn to imitate the song of a tutor. Vocal learning is separated into distinct phases, during which the learner needs to be exposed to specific stimuli to enable learning. If in the critical period exposure is prevented, learning fails. For songbirds, there is a critical period in which the young bird has to be exposed to the tutor song, which it is supposed to learn. After this critical period has passed no more songs can be learned. After the student bird has learned a sensory representation of the tutor song in the sensory period, in a different phase it performs sensorimotor learning, where it needs to be exposed to its own vocalizations. In this phase, also called a babbling phase, the young bird generates highly variable sounds of low amplitude similar to babbling behaviour in infants. From these small sound snippets, or subsong, later an imitation of the whole song emerges. This song will first still be variable and therefore be called plastic song and then crystallize into the adult version.

2.2.2 Basic Neuroanatomy

The avian song system is a complex structure (see figure 2.2.2), of which I will highlight some key aspects here, that are important to understand this thesis. For more details refer to [BD13] or [RPMJ04].

In the avian brain, sounds are processed in several auditory areas (caudal mesopallium (CM), medial part of the dorsolateral thalamic nucleus(DLM), nucleus avalanche (Av), caudomedial nidopallium (NCM), nucleus interface (NIf), nucleus ovoidalis (OV), nucleus uvaeformis (Uva), field L), which I will not discuss in detail, because they are not part of the song system. These areas project to the motor part of the song system, specifically to HVC (abbreviation used as a proper name, formerly higher vocal center), which is a premotor area. Activations in HVC are extremely sparse and involved in the

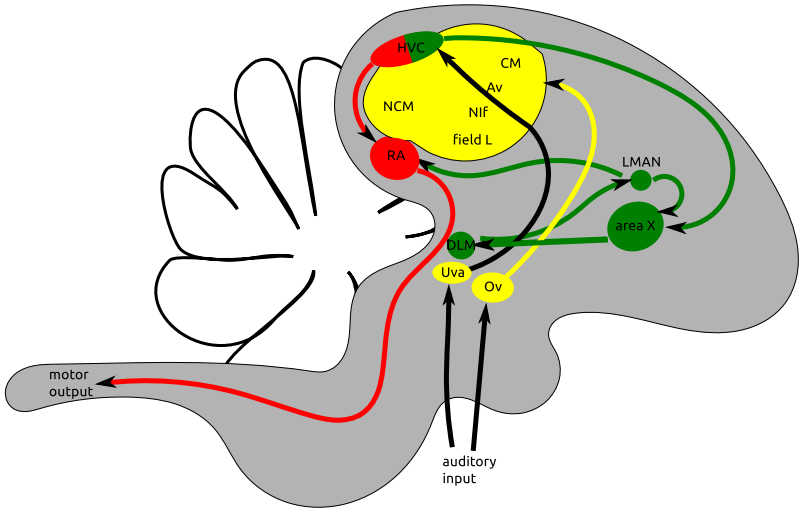


Figure 2.2: Schematic drawing of the avian brain. Processed auditory input is fed into premotor area HVC, which innervates RA, which in turn controls respiratory centers and the motor neurons controlling the vocal organ. LMAN provides further input into RA.

timing and sequencing of the song [HKF02, FS10]. HVC innervates RA (robust nucleus of the arcopallium), which directly controls the motor neurons in the vocal organ and the respiration. Both, HVC and RA are important for song production throughout life [NSL76]. Lesions cause song disruption or even muteness. HVC and RA are also indirectly linked via the anterior forebrain pathway (AFP), which is crucial for song learning and the slight variability present in adult song [KB06, TBW$^+$11]. In particular, area LMAN (lateral magnocellular nucleus of the anterior nidopallium) in the AFP has been viewed to be particularly important for learning in songbirds, since it provides highly variable input into RA during learning, in particular in the sensorimotor phase, and comparatively stereotyped activations during directed singing in the adult bird.

2.2.3 Mirror Neurons

Mirror neurons were originally discovered in the frontal cortex of monkeys [DPFF$^+$92]. These are neurons that are active during the performance of an act and in a similar way during the passive observation of that same act. Neurons with this general property were later discovered in songbirds as well. I will here only provide a short introduction, for a review see [Moo14].

During the vocalization of a songbird, certain neurons in areas HVC, RA and LMAN are active. In sleeping or anaesthetized birds, these same neurons are also active during passive replay of these songs. In fact, in zebra finches, all areas downstream of HVC are activated during playback. However, the most striking observations were made with chronic extracellular recordings in freely singing swamp sparrows: Here, the activations in HVC neurons innervating area X are active at almost precisely the same time relative to the song during singing as during replay [PPNM08]. It would be easy to assume that this effect is just a side-effect of the reaction of these HVC neurons to identical acoustic stimuli.

However, it was found that the singing related activity of these cells is not perturbed by altered auditory feedback during singing [PPNM08, HTY$^+$14]. Furthermore, the precise mirroring activations can only be found in sleeping or anaesthetized birds, which implies that sensory input into HVC is gated off during singing.
While in HVC$_X$ neurons mirroring activity with a zero time delay between vocalization and playback were found, neurons in LMAN display mirroring behaviour with a time lag of about $40 - 60ms$ [GKGH14]. In their study, Giret et al. observed singing behaviour in freely moving zebra finches and playback during sleep. In their study, they also estimate the loop delay, that is the time it takes for motor activation in LMAN to elicit sound and the time that it takes for a response in LMAN to occur during playback of the song. They estimated the loop delay to lie between the minimal measured value at $32ms$ and the median loop delay at $56ms$. The delay of the mirroring activity in LMAN thus roughly matches the loop delay. Throughout this thesis, I will therefore assume a loop delay in the order of magnitude of $40ms$.

Chapter 3

Theoretical Background

Since the biology inspiring theoretical investigations of learning is very complex, any theoretical approach will have to be based on simplifications of all components of the system. In particular, if, like here, a modular system of interacting modules is to be investigated, the elements of which the modules are comprised should be chosen to be as simple as possible to allow for reasonable computing times. In this chapter, I will provide an overview about the standard neuron models that will be used throughout this work, as well as the ways in which these neurons then are coupled to form networks. I will go on to discuss different standardized learning problems, as well as a number of learning rules that strive to solve these problems. The type of learning problems discussed in this chapter usually consist of a desired input-output relationship in a given neuronal network, which is supposed to be acquired during training. During training, an outside entity gives feedback to the network on how well it is doing (reinforcement learning, 3.4.3) or even how to change the connections from input to output to derive the required input-output relationship (supervised learning 3.4.2). Most of this description loosely follows [DA01].

3.1 Neuron Models

Due to the complex nature of real biological neurons, neuron models always have to make some simplifications. Which simplifications are reasonable depends on the context, in which the models will be used, since there is a tradeoff between biological realism on the one hand and computational cost and analytical tractability on the other hand. While there are very complex neuron models that capture many different aspects of neurons, here, I will limit the description to the basic neuron models that are widely used in modelling studies.
All these neuron models have two things in common: The spatial structure of the neuron is omitted, which leads to point neuron models. Furthermore, all model neurons reproduce the basic non-linearity of the spiking process. This spiking process is modelled in great detail in the Hodgkin-Huxley type model (see section 3.1.2.3) or just as a threshold crossing event in the (conductance-based) integrate-and-fire neuron (see section 3.1.2.1 and 3.1.2.2).

3.1.1 Rate Neurons

Rate neuron models are models that operate on the firing rate rather than on single spikes. Here, I will only present the very basic idea of rate neurons, since they will not be used in this thesis. However, I want to give a very basic introduction into what a rate neuron is and how it operates, because rate neurons have been used in some of the discussed literature.
There are many different rate neuron models, which have in common that the neuron receives input from its presynaptic neurons and converts it to an output. Both, the input h_i and the output y_i are modelled as real numbers. The output is given by an activation function $g(h_i)$, such that

$$y_i = g(h_i) = g\left(\sum_j w_{ij}x_j + h_i^{ext}\right) \tag{3.1}$$

where w_{ij} is the synaptic weight between presynaptic neuron j and postsynaptic neuron i. x_j is the activation of presynaptic neuron j, h_i^{ext} is an external, non-synaptic input into neuron i.
The activation function $g(h)$ can be chosen in different ways to capture different aspects of the neurons dynamics. Common choices are the logistic function, the rectifying bracket ($[h] = h$ if $h > 0$ and 0 otherwise) and the Heaviside step function

$$\Theta(h) = \begin{cases} 1 & \text{if } h > 0 \\ 0 & \text{else} \end{cases} \tag{3.2}$$

Rate neurons can be modelled with or without an explicit dependence on time. In the former case, the firing rate depends on time via the time dependent external or synaptic input. In the latter case, the input is assumed to be stationary at least for some periods of time, which simplifies the model significantly and can be justified in some contexts. These neuron models are often used to investigate the propagation of activity through layered networks.

3.1.2 Spiking Neurons

For a detailed description and discussion of how different spiking neuron models can be reduced to each other see [AK90].

3.1.2.1 The Integrate-and-Fire Neuron

The simplest model that captures the dynamics of the membrane potential $V(t)$ is the leaky integrate-and-fire neuron. The basic assumption is that the subthreshold behaviour of the neuron can be modelled as a capacitor with capacitance C_m. It can be charged by external input I_e, generating a potential different from the resting potential V_{rest}, to which the voltage decays with time constant $\tau_m = C_m \cdot R_m$, when no further input is given. R_m is the resistance modelling the leak currents through the membrane. The voltage is then given by

$$\tau_m \frac{\mathrm{d}V}{\mathrm{d}t} = -V + V_{rest} + R_m I_e \tag{3.3}$$

Throughout this thesis, the resting potential V_{rest} is set to zero without loss of generality. Furthermore, in the following, it will be assumed that $R = 1$, since the resistance can always be absorbed in the description of the input current I_e. There are synaptic inputs from afferent neurons I_{syn} and external currents I_{ext} that are given by an experimentator via an electrode, which constitute the full input current $I_e = I_{syn} + I_{ext}$.
Whenever the membrane potential V crosses the spiking threshold V_{thresh} at time t_k, it is reset to a reset potential $V_{reset} \leq V_{rest} < V_{thresh}$, which generally is at or below the resting potential. At this point in time, a spike is registered. If the reset potential is below the resting potential, the neuron is in a hyperpolarized state after each spike, thus hindering spiking again immediately after a spike. Spikes are generally modelled to be delta pulses, such that the spike train of the neuron s is given by

$$s(t) = \sum_k \delta(t - t_k) \tag{3.4}$$

where t_k is the time of the kth spike.
The synaptic currents I_{syn} can be modelled in different ways. Assuming a very fast opening and closing of the ion channels in the synapse, the synaptic current can be modelled as a sum of delta functions of height of the synaptic weight w_i between a presynaptic neuron i and the postsynaptic neuron that is being modelled:

$$I_{syn} = \sum_i \sum_k w_i \delta(t - t_k^i - \tau_a - \tau_d) \tag{3.5}$$

where τ_a is the time that the signal travels from the soma of the presynaptic neuron to the synapse and τ_d is the time that the signal travels from the synapse to the soma of the postsynaptic neuron. These delays are often modelled as a single delay of synaptic transmission τ.
This model of synaptic transmission generates postsynaptic potentials (PSPs) of the shape of an exponential decay:

$$V_\delta(t) = \Theta(t) \exp\left(-\frac{t}{\tau_m}\right) \tag{3.6}$$

where it is assumed that the input into the postsynaptic neuron happens at $t = 0$. $\Theta(t)$ is the Heaviside step function given by equation 3.2. If the closing process of the ion channels in the synapse is modelled to take a finite time τ_{syn}, the synaptic current can be modelled as a sum of step-and-decay functions, that rise with the synaptic weight w_i between a presynaptic neuron i and the postsynaptic neuron that is being modelled:

$$\tau_{syn}\dot{I}_{syn} = -I_{syn} + \sum_i \sum_k w_i \delta(t - t_k^i - \tau) \tag{3.7}$$

The shape of the resulting PSP is then given by the convolution of this synaptic input current with the response kernel of the neuron to delta shaped input given by equation 3.6. The shape of the membrane potential in response to a single presynaptic spike in neuron i of synaptic weight w_i at time $t = 0$ is then given by

$$V_{t_s} = \frac{1}{\tau_m - \tau_s}\left(\exp\left(-\frac{t}{t_m}\right) - \exp\left(-\frac{t}{t_s}\right)\right)\Theta(t) \tag{3.8}$$

Due to the linearity of the differential equation governing the voltage V, it is possible to reformulate the integrate-and-fire neuron into the Spike Response Model [GK02]. The

membrane voltage $V(t)$ is then given by the sum of weighted synaptic input kernels $\varepsilon(s)$ (postsynaptic potentials, PSPs) and reset kernels $R(s)$, which model the neuronal reset after a spike. External input currents $I_{ext}(t)$ are low-pass filtered with a response kernel $\kappa(s)$. The full equation reads

$$V(t) = \sum_i w_i \sum_k \varepsilon(t - t_k^i - t_{delay}) + \sum_{t_j} R(t - t_j) + \int_0^\infty \kappa(t-s) I_{ext}(s) ds. \tag{3.9}$$

Here, w_i is the weight from presynaptic neuron j to the postsynaptic neuron. $\kappa = \exp\left(-(t-s)/\tau_m\right)$ is the passive response kernel by which external currents are filtered. The other kernels are

$$\begin{aligned} \varepsilon(s) &= \Theta(s) \frac{1}{\tau_m - \tau_s} \left(\exp(-s/\tau_m) - \exp(-s/\tau_s)\right) \\ R(s) &= \Theta(s)(V_{reset} - V_{thr}) \exp(-s/\tau_m). \end{aligned} \tag{3.10}$$

This formulation can be useful in the derivation of learning rules based on the dynamics of the neuron model.

3.1.2.2 The conductance-based Integrate-and-Fire Neuron

To increase the biological accuracy of the above model, it is possible to separate inhibitory from excitatory inputs. The presynaptic population is split into N_{ex} excitatory and N_{in} inhibitory neurons. The postsynaptic neuron can then be modelled as a conductance based LIF neuron governed by

$$C_m \frac{\mathrm{d}V}{\mathrm{d}t} = -g_L(V - V_L) - (g_{sl} + g_f)(V - V_h) - g_{ex}(V - V_{ex}) - g_{in}(V - Vin) \ , \tag{3.11}$$

where V denotes the membrane potential, C_m the membrane capacitance, V_L the resting potential, g_L the leak conductance, V_i and V_{ex} the reversal potential of inhibition and excitation, respectively and g_{in} and g_{ex} their respective conductances. The spike after-hyperpolarisation is modelled to be biphasic, consisting of a fast and a slow part, described by conductances g_f and g_{sl} that keep the membrane potential close to the hyperpolarisation potential $V_h = V_i$. When the membrane potential surpasses the spiking threshhold V_{thr} at time t_{post}, a spike is registered and the membrane potential is reset to $V_{reset} = V_h$. All conductances are modelled as step and decay functions. The reset conductances are given by

$$\tau_{f,sl} \dot{g}_{f,sl} = -g_{f,sl} + \Delta g_{f,sl} \sum_{t_{post}} \delta\left(t - t_{post}\right) \ , \tag{3.12}$$

where Δg_{sl} resp. Δg_{sl} is the increase of the fast and slow conductance at the time of each postsynaptic spike. They decay back with time constants $\tau_f = \tau_s < \tau_{sl} = C_m/g_L$. The input conductances g_{ex} and g_{in} are step and decay functions as well, that are increased by w_i when presynaptic neuron i spikes and decay with time constant τ_s. w_i denotes the strength of synapse i.

3.1.2.3 The Hodgekin-Huxley Neuron

The Hodgkin-Huxley model is a sophisticated neuron model, where the flow of sodium and potassium ions through voltage dependent channels in the membrane are explicitly

modelled. Additionally, there is a leak conductance, which models other ionic currents. The full equation for the voltage is given by

$$C_m\dot{V} = -g_L(V - V_L) - g_K n^4(V - V_K) - g_{Na}m^3h(V - V_{Na}) - g_{ex}(V - V_{ex}) - g_{in}(V - Vin) \tag{3.13}$$

where $V_L = -65mV$ is the leak potential, $g_L = 0.1mS/cm^2$ is the leak conductance, g_{ex} resp. g_{in} are the conductance governing excitatory resp. inhibitory input from the input populations and $V_{ex} = 0mV$ resp. $V_{in} = -75mV$ are their reversal potentials. $V_{Na} = 55mV$ is the reversal potential of sodium, $V_K = -90mV$ is the reversal potential of potassium, $C_m = 1\mu F/cm^2$ is the membrane capacitance, $g_K = 9mS/cm^2$ is the maximum potassium conductance and $g_{Na} = 35mS/cm^2$ is the maximum sodium conductance.
The conductance variables $n(t, V)$, $m(t, V)$ and $h(t, V)$ are time and voltage dependent and take values between 0 and 1. They are given by

$$\dot{n} = \alpha_n(V)(1 - n) - \beta_n(V)n \tag{3.14}$$
$$\dot{m} = \alpha_m(V)(1 - m) - \beta_m(V)m \tag{3.15}$$
$$\dot{h} = \alpha_h(V)(1 - h) - \beta_h(V)h \tag{3.16}$$

where

$$\alpha_n(V) = \frac{(-0.01(V(t) + 55))}{(\exp(-0.1(V + 55)) - 1)} \tag{3.17}$$
$$\beta_n(V) = 0.125\exp(-\frac{(V(t) + 65)}{80}) \tag{3.18}$$
$$\alpha_m(V) = \frac{(-0.1(V + 40))}{(exp(-0.1(V + 40)) - 1)} \tag{3.19}$$
$$\beta_m(V) = 4\exp(-0.0556(V + 68)) \tag{3.20}$$
$$\alpha_h(V) = 0.07\exp(-\frac{(V + 65)}{20}) \tag{3.21}$$
$$\beta_h(V) = \frac{1}{exp(-0.1(V + 35)) + 1} \tag{3.22}$$

To facilitate reading, units are dopped; voltages are in mV, time is in ms. Parameters are fitted to experimental data and taken from [AK90].
When a sufficient input current is given to the neuron, the membrane potential rises over a critical value, which causes a rapid increase in the variable m, which activates a positive inward sodium current. The membrane potential rises sharply, until the variables h and n adjust to the change in membrane potential. The variable h decreases, thus decreasing the sodium current that caused the rise in the membrane potential. At the same time, n increases and causes a positive outward potassium current which drives the membrane potential below resting potential into hyperpolarization. Over time, all conductance variables m, h and n return to their resting values.
The parameters of the model can be tweaked to provide a more or less strong hyperpolarization after each spike.

3.2 Poissonian Spiking and Noise

The spiking neuron models presented above provide highly reliable responses to defined outputs, similar to the response of a real neuron to a defined input current [MS95]. How-

ever, in vivo studies show that spiking in real neuronal networks is highly irregular. The statistics of the spiking can be well described as poissonian processes. The probability of measuring n spikes in a time interval T is given by

$$P(n) = \frac{(rT)^n}{n!} \exp(-rT) \tag{3.23}$$

if the underlying process has a firing rate r. Here, the firing rate is assumed to be constant during the time interval T. However, this is not always the case, such that it is useful to define an instantaneous firing rate $r(t)$, which in modelling studies may depend on the voltage.
Transforming a firing rate into a set of discrete spiking events for simulation purposes can be done via

$$P(\text{spike between } t \text{ and } t + \Delta t) = r(t)\Delta t \tag{3.24}$$

where Δt is the bin size of the time binning. It should be chosen small enough, such that the probability of having more than one spike per interval is very small, i.e. $r(t)\Delta t << 1$.
In modelling studies, it is often useful to explicitly model noise. In spiking neurons that can be achieved by an additional noise input current I_{noise}, which is often chosen as gaussian white noise centered around zero.
However, for analytical studies it can be simpler to assume that the spike generation process is stochastic. To that end, for example in the SRM_0 with exponential escape noise [GK02], a conversion of the membrane potential to a instantaneous firing rate is defined by

$$r(t) = r_0 \exp\left(\frac{V(t) - V_{thr}}{\Delta V}\right) \tag{3.25}$$

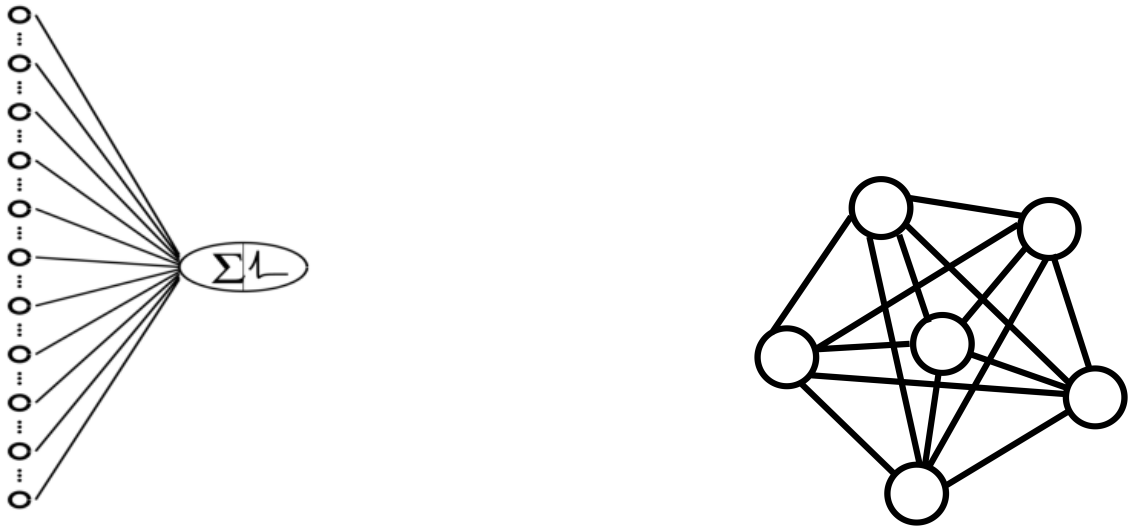
ΔV describes how strong the influence of stochastic spike generation is. If ΔV is chosen large, spiking is relatively independent from the voltage and therefore, very stochastic. However, if ΔV is chosen very small, the influence of the voltage is bigger and spiking becomes more predictable.

3.3 Network Models

The brain consists of interconnected neurons. The architecture of these connections has a relevant contribution to the computational output. In modelling studies, different architectures are considered in different network models.

3.3.1 Feed-Forward Networks

In feed-forward networks, neurons are organized in layers. A neuron i in a layer receives input I from a neuron j in the previous layer, its input layer, whenever j is active (see figure 3.1(a) for a sketch). The activity of neuron j is denoted in a_j. The neurons from the input layer are connected to the output neurons via weights w_{ij}, which signify the strength of the connection. Then the input into each neuron is just the summed output from affarent neurons, weighted by their respective weights. This input will be converted to output either by an activation function (see section 3.1.1) or given as input current into



(a) (b)

Figure 3.1: Sketch of different network setups. (a) Feed-forward network: An output neuron receives directed input from an input layer, but the activation of the output neuron has no influence on the input layer. (b) Recurrent network: All neurons are interconnected, such that the activation of each neuron influences its own future activation via the network activation.

the spiking neuron model (see e.g. section 3.1.2.1).
There is no feedback from the output layer back to the input layer, which is why these networks are called feed-forward networks. These networks are widely used in classification problems (see section 3.4.2.1 and section 3.4.2.3).

3.3.2 Recurrent Networks

By contrast, in recurrent networks generally all neurons are interconnected. This means, in principle, each neuron receives input from all other neurons or a subset of neurons, but crucially, the output of each neuron is fed back into the network and thus implicitly to itself.
Because of the connectivity, the weights w_{ij} can be organized into a weight matrix w, where the column indexes the presynaptic neuron and the row indexes the postsynaptic neuron. The connections can have a delay τ.
A peculiar property of recurrent networks is that they are highly sensitive to noise. If sequences of spike patterns in recurrent networks are considered, a slight difference in the pattern at time t can have a large influence on the later activation patterns.

3.4 Learning

In artificial neuronal networks, neurons are interconnected by synaptic weights, which define the behaviour of the network in response to a given input. These weights are a simple model of synaptic connections between biological neurons. The underlying hypothesis in

neuroscience is that synaptic plasticity is the neuronal substrate for learning and memory formation. Physiological studies have shown that synapses change in response to the pairwise activity of the connected neurons. In real neurons, the synaptic strength can change on a diversity of time scales. In this thesis, only long term plasticity is treated, which in experiments takes seconds to minutes to induce and express, and stays at least for one hour (presumably longer).
In modelling, there are two different approaches to synaptic learning: One is the top down approach, where a goal is defined and then an update rule for synaptic weights is derived to achieve that goal. These update rules often rely on a teaching signal giving explicit information about the necessary synaptic changes to the neuron. This type of learning is called supervised learning. The other approach, the bottom up approach, starts from experimental data. The learning behaviour of biological synapses is observed and modelled. Then, these simplified models can be applied to neuronal networks to see, if they are useful for computational purposes. This approach results in unsupervised learning.

3.4.1 Unsupervised Learning

Due to the current state of experimental techniques, the synaptic change in biological studies is a function of the pairwise activity. The experimental results can be subsumed in mathematical models and implemented in artificial neuronal networks to analyze their consequences. The learning system evolves according to the learning rule without supervision from an outside entity or teacher. This type of learning is likely to happen in real biological neurons. Here, two very simple activity dependent plasticity rules will be introduced that are widely used in modelling studies: Hebbian learning and spike pair spike timing dependent plasticity.

3.4.1.1 Hebbian Learning Rule

The idea of Hebbian learning is derived from the observation that co-activation of neurons leads to a synaptic strengthening. This is formalized in the Hebbian learning rule to train Hopfield networks, which are a modelling attempt for content-addressable memory and associative learning [Hop82, Hop07] (see section 3.4.3.1.1). This is the most simple form of an activity dependent learning rule.
Following the description given in [HKP91], x_i is the activation of neuron i and x_j is the activation of neuron j. Then the weight w_{ij} between them as defined by Hebbian learning is given by

$$\Delta w_{ij} \propto x_i x_j \tag{3.26}$$

It is possible using this weight change to make defined activity patterns the attractors of the dynamics of recurrent networks of rate neurons (Hopfield-networks, see section 3.4.3.1.1). The downside is that this rule leads to symmetric weights, which is neither biologically realistic, nor computationally efficient. More sophisticated learning rules thus try to include more information about the activation patterns than just co-occurrence. However, it is possible to imprint desired activation patterns onto Hopfield networks (see below).

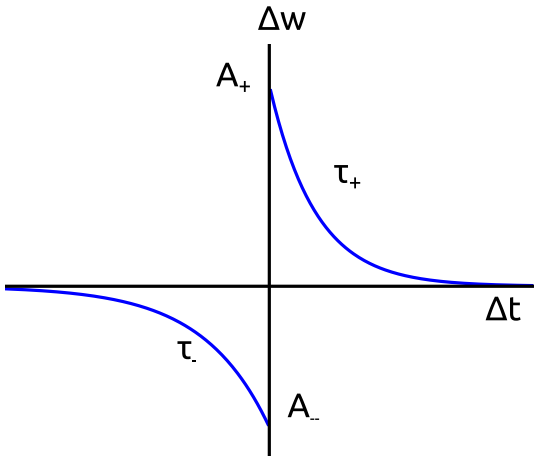


Figure 3.2: The STDP window as a result of the simple spike pair STDP model: The weight change decays exponentially with τ_+ resp. τ_- for large time distances between pre- and postsynaptic spiking. The amplitude of the weight change is given by A_+ resp. A_-.

3.4.1.2 Spike Pair Spike Timing Dependent Plasticity

Spike timing dependent plasticity (STDP) has been observed experimentally, often in the classical, hebbian form introduced here (see 2.1.3.1 for details on experimental findings). In experiments, it is often studied in a setting, where the synaptic change upon pairwise stimulation of the pre- and postsynaptic neuron in dependency of the timing difference between these stimulations is observed. In typical experiments on synapses of excitatory neurons, STDP is observed in its standard form: a postsynaptic spike which arrives at the synapse shortly before a presynaptic spike leads to synaptic depression, while a presynaptic spike arriving at the synapse shortly before the postsynaptic spike leads to synaptic potentiation. This sensitivity of the learning rule to the specific order of spiking makes it a possible mechanism for learning timing dependent responses.
Let us consider a single spike pair where t_{pre} is the time of the presynaptic spike and t_{post} is the time of the postsynaptic spike. Then $\Delta t = t_{post} - t_{pre}$ is the time distance between the presynaptic spike and the postsynaptic spike. This allows to formulate a simple description of the weight change according to spike pair STDP for this spike pair as

$$\Delta w_1^{spSTDP} = \begin{cases} A_+ \exp\left(\frac{-\Delta t}{\tau_+}\right) & \text{if } \Delta t > 0 \\ A_- \exp\left(\frac{\Delta t}{\tau_-}\right) & \text{else} \end{cases} \tag{3.27}$$

where A_+, A_-, τ_+ and τ_- are the parameters of the model, which define the shape of the STDP window (see figure 6.1).

The exponential shape of the STDP window captures the experimental results on STDP relatively well. To generalize the model to arbitrary spike patterns, in additive spike pair STDP, the resulting weight changes are simply superposed, such that the total weight change is given by

$$\Delta w^{spSTDP} = \sum_{\Delta t} \Delta w_1^{spSTDP} \tag{3.28}$$

Because of its simplicity, spike pair STDP is widely used in modelling studies. However, more sophisticated experiments show that the spike pair STDP model likely is an oversimplification of real synapes [FD02, WGNB05, SH06, AWP13].

3.4.2 Supervised Learning

Supervised learning is used to imprint a precise set of desired input-output relationships onto neuronal networks. If the output neuron is required to respond to certain classes of input patterns with a specific output activity, this process is called classification. During the learning process, a supervisor or teacher provides detailed feedback to the synaptic connections on the success of learning and the nature of the synaptic changes necessary to achieve the desired input-output relationship. The formulation of the input-output relations depends on the network structure and neuron type: In the most simple setting in a feed-forward network, an output neuron is required to respond to a given spatial input pattern either with positive activity or not. This is called the Perceptron problem and will be discussed in the next section. The Perceptron setting can be extended to the temporal domain with spiking neurons, the Tempotron: Here, the output neuron is required to classify two sets of spatio-temporal input patterns by either spiking or not spiking. The natural extension of the Tempotron is the Chronotron: Here, the output neuron is required to respond to a spatio-temporal input spike pattern with a spike at a precisely defined time. This problem and its solutions will be discussed in section 3.4.2.3. Supervised learning is also applied to learning in recurrent networks. Here, either a stationary activation pattern in response to a noise or partial input of that pattern or even elongated spatio-temporal activation patterns are the goal of learning. Due to the high sensitivity of activations in recurrent networks to noise, it is difficult to imprint stable activation sequences.

3.4.2.1 The Perceptron

The Perceptron is a toy model of a simple feed-forward neural network, that can learn to distinguish two different classes of inputs. To that end, it is required to respond to one class of input patterns with activation and to the other class with non-activation. This can also be viewed as a case of associative learning, where the input pattern is associated with a given output.
Consider a simple feed-forward network, which consists of a layer of N input neurons and a single output neuron that is trained to perform the desired classification. Figure 3.1(a) shows an example of such a network. In the original Perceptron setup as introduced in [HKP91], the state of each input neuron is called ξ_i and takes the values $\xi_i \in \{-1, 1\}$. The output neuron takes states $O \in \{-1, 1\}$. It is a simple threshold unit, i.e. it computes its output according to

$$O = g(h) = g\left(\sum_i w_i \xi_i\right) \tag{3.29}$$

where w_i is the connections strength from input neuron i to the output neuron and $g(h)$ is the activation function. In the simple case of deterministic threshold units it is just the sign function:

$$g(h) = \mathrm{sgn}(h) \tag{3.30}$$

3.4.2.1.1 The Original Perceptron Learning Rule

In simple classification problems, there are two classes of μ input patterns, which are supposed to be distinguished by the output neuron. The input vectors of pattern μ will

be denoted ξ^μ. This implies that for each input pattern ξ^μ there is a desired output ζ^μ. The goal then is that

$$O^\mu \stackrel{!}{=} \zeta^\mu \ . \tag{3.31}$$

Since O^μ is given by eq. (3.29), the weight w_i have to be chosen such that the summed and weighted input into the output neuron is either positive or negative.
It is possible to interpret eq. (3.29) as a scalar product, which makes it possible to rewrite it as

$$\zeta^\mu \stackrel{!}{=} O^\mu = \mathrm{sgn}\left(\vec{w} \cdot \vec{\xi}^\mu\right). \tag{3.32}$$

This shows that the output is just the sign of the input projected onto the weight vector. Therefore, the boundary between positive and negative output is given by the plane defined by $\vec{w}\vec{\xi}^\mu = 0$ through the origin and perpendicular to $\vec{w}$.
The goal here, is to choose $\vec{w}$ such that this plane separates patterns with desired positive output ($\zeta^\mu = 1$) from patterns with desired negative output ($\zeta^\mu = -1$).
This is not always possible, since the patterns may not always be linearly separable, that is there may be two or more patterns that require synaptic weights that are incompatible to provide correct output.
For robustness to noise in the input patterns, it is useful to define a margin κ, which defines a minimum distance between the input into the output neuron and zero, such that

$$\zeta^\mu h^\mu > N\kappa \tag{3.33}$$

The original Perceptron learning rule is then given by

$$\Delta w_i = \eta\Theta\left(N\kappa - \zeta^\mu h^\mu\right)\zeta^\mu \xi_i^\mu \tag{3.34}$$

This quantity is only larger than zero, if the condition in equation 3.33 is not fulfilled, that is the output is different from the required output. Then the weight from input neuron i to the output neuron is increased, if the activation in both has the same sign and decreased, if they are of different sign. Hence, the weight change moves the output in the desired direction. Due to the constant size of the weight change in each step, this learning rule converges in a finite number of steps, if learning is possible (see [HKP91] for a proof). Furthermore, learning stops, when the actual output is the desired output, such that overlearning due to repeated presentation of the input patterns is not possible. These are highly desirable qualities in a learning rule.

3.4.2.2 The Tempotron

The natural extension of the Perceptron problem is the Tempotron. Here, the output neuron is taught to classify elongated spatio-temporal patterns and respond to them with either a spike or no spike. This concept was introduced in [GS06], where it was found that in principle, integrate-and-fire neurons are capable of learning such spike timing based decisions, since the state of the neuron is dependent on the order of inputs into a neuron.

3.4.2.3 The Chronotron

The Chronotron problem is the extension of the Tempotron problem from the pure classification task on spatio-temporal patterns: The output neuron is required to provide one

output spike at a precisely defined time during each input pattern.
There are several learning rules that attempt to solve these classification problems, some of which I will introduce here.

3.4.2.3.1 The δ-rule and ReSuMe.

The δ-rule, also called the Widrow-Hoff-rule [HKP91], lies at the core of a whole class of learning rules used to teach a neuronal network some target activity pattern. Synaptic changes are driven by the difference of desired and actual output, weighted by the presynaptic activity:

$$\Delta w(t) \propto f_{pre}(t) \left(f_{post}^{target}(t) - f_{post}^{actual}(t) \right) \ . \tag{3.35}$$

Pre- and postsynaptic firing rate are denote $f_{pre,post}$. The target activity $f_{post}^{target}(t)$ is some arbitrary time dependent firing rate. The actual self-generated activity $f_{post}^{actual}(t)$ is given by the current input or voltage of the postsynaptic neuron (depending on the formulation), transformed by the input-output function $g(h)$ of the neuron.
ReSuMe (short for Remote Supervised Method) is a supervised spike-based learning rule first proposed in 2005 [PK10]. It is derived from the Widrow-Hoff rule for rate-based neurons, applied to deterministic spiking neurons. Therefore, continuous time dependent firing rates are replaced by discrete spiking events in time, expressed as sums of delta-functions. Because these functions have zero width in time, it is necessary to temporally spread out presynaptic spikes by convolving the presynaptic spike train with a temporal kernel. Although the choice of the kernel is free, usually a causal exponential kernel works best. The weight change is given by

$$\dot{w}(t) \propto [S_d(t) - S_o(t)] \left[a_d + \int_0^\infty \exp(-s/\tau_{plas}) S_i(t-s) ds \right] \ , \tag{3.36}$$

where $S_d(t)$ is the desired, $S_o(t)$ is the self-generated and $S_i(t)$ the input spike train at synapse i. τ_{plas} is the decay time constant of the exponential kernel. a_d is a constant which makes sure that the actual and target firing rates match; learning also works without. ReSuMe converges when both actual and desired spike lie at the same time, because in this case the weight changes exactly cancel each other out.

3.4.2.3.2 E-Learning.

E-Learning was conceived as an improved learning algorithm for spike time learning [Flo12a]. It is derived from the Victor-Pupura distance (VP distance) between spike trains [VP96]. The VP-distance is used to compare the similarity between two different spike trains (see section 3.5.2).
E-Learning is a gradient descent on the VP distance and has smoother convergence than ReSuMe. In this rule, first the actual output spike train is compared to the desired spike train. The VP algorithm determines if output spikes must be shifted or erased or if some desired output spike has no close actual spike so a new spike has to be inserted. Based on this evaluation, actual and desired spikes are put in three categories:

- Actual output spikes are "paired" if they have a pendant, i.e. a desired spike close in time and no other actual output spike closer (and vice versa). These spikes are put into a set S.

- Unpaired actual output spikes that need to be deleted are put into the set D.
- Unpaired desired output spike times are put into the set J, i.e. the set of spikes that have to be inserted.

To clarify, S contains pairs of "paired" actual and desired spike times, D contains the times of all unpaired actual spikes, and J the times of unpaired desired spike times. With the PSP sum as above, the E-Learning rule is then

$$\Delta w_i = \gamma \left[\sum_{t_{ins} \in J} \lambda_i(t^{ins}) - \sum_{t^{del} \in D} \lambda_i(t^{del}) + \frac{\gamma_r}{\tau_q^2} \sum_{(t^{act}, t^{des}) \in S} (t^{act} - t^{des}) \lambda_i(t^{act}) \right] . \quad (3.37)$$

γ is the learning rate, and γ_r is a factor to scale spike shifting relative to deletion and insertion.
The former two terms of the rule correspond to ReSuMe, except the kernel is not a simple exponential decay. The advantage of E-Learning is that the weight changes for spikes close to their desired location are scaled with the distance, which improves convergence and consequentially memory capacity.

3.4.2.3.3 FP-Learning.

FP-Learning [MRÖS14] was devised to remedy a central problem in learning rules like ReSuMe and others. Any erroneous or missing spike "distorts" the time course of the membrane potential behind it compared to the desired final state. This creates a wrong environment for the learning rule, and weight changes can potentially be wrong. Therefore, the FP-Learning algorithm stops the learning trial as soon as it encounters any spike output error. Additionally, FP-Learning introduces a margin of tolerable error for the desired output spikes. An actual output spike should be generated in the window of tolerance $[t_d - \epsilon, t_d + \epsilon]$ with the adjustable margin ϵ. Weights are changed on two occasions:

1. If a spike occurs outside the window of tolerance for any t_d at time t_{err}, then weights are depressed by $\Delta w_i \propto -\lambda_i(t_{err})$. This also applies if the spike in question is the second one within a given tolerance window.
2. If $t = t_d + \varepsilon$ and no spike has occured in the window of tolerance, then $t_{err} = t_d + \varepsilon$ and $\Delta w_i \propto \lambda_i(t_{err})$.

In both cases, the learning trial immediately ends, to prevent that the "distorted" membrane potential leads to spurious weight changes. Because of this property, this rule is also referred to as "First Error Learning".

3.4.3 Reinforcement Learning

Reinforcement learning is a special case of supervised learning, where the teacher does not give specific information about the changes that need to be made to the synaptic connections in order to accomplish the desired task. Instead it only provides a feedback on how well the task is performed in the form of a reward or a punishment.
Reinforcement learning can be interpreted as trial-and-error learning. The learning system is active in a particular way and then this activity is evaluated by the critic, that gives a (possibly negative) reward. Next, the system tries out a change and if the received reward

is higher, it will keep the change and otherwise discard it.

In this way, e.g. irregularly spiking neural networks can estimate the gradient of the reward signal and perform a stochastic gradient ascent on the expected reward [XS04]. In their model, Xie and Seung introduce stochastic spiking neurons i, which receive input from input neurons j. The strength of the connection between j and i is given by the synaptic weight w_{ij}. The input current into neuron i then given by

$$I_i(t) = \sum_j w_{ij} h_{ij}(t) \tag{3.38}$$

where h_{ij} evolves according to

$$\tau_s \frac{\mathrm{d}h_{ij}}{\mathrm{d}t} + h_{ij} = \sum_a \delta(t - T_j^a)\zeta_{ij}^a \tag{3.39}$$

where ζ_{ij}^a is a binary random variable modelling the stochastic nature of synaptic transmission and T_j^a is the time of the ath spike in input neuron j. The input current $I_i(t)$ is converted to an instantaneous firing rate $\lambda_i(t)$ by

$$\lambda_i(t) = f_i(I_i(t)) \tag{3.40}$$

where f_i is the current-discharge relationship.
The learning rule is then given by

$$\Delta w_{ij} = \eta R e_{ij} \tag{3.41}$$

where R is a reward signal, η is a learning rate and e_{ij} is an eligibility trace given by

$$e_{ij} = \int_0^T \mathrm{d}t \Phi_i(I_i) \left[s_i(t) - f_i(I_i) \right] h_{ij} \tag{3.42}$$

where T is the length of the learning episode and $s_i(t) = \sum_a \delta(t - T_i^a)$ is the spike train in neuron i. $\Phi_i(I_i)$ is a function that scales the weight changes depending on the current firing rate. Learning works, because if the actual activation of the neuron is larger than the instantanous firing rate and this is rewarded ($R > 0$), then weights from positively contributing input neurons are increased to increase the instantaneous firing rate upon repetition. Similarly, if the actual output of the neuron is above the instantaneous firing rate and this is punished ($R < 0$), weights are changed to decrease the firing rate. The same mechanism also applies for an actual activation below the firing rate, such that the firing rate is changed in the desired direction. By this learning mechanism, a gradient ascent on the reward is performed.
For constant reward and an output spike train that is enforced by a teacher, this is very similar to the δ-rule.

In a study by Fiete et al., reinforcement learning has been shown to be applicable to more realistic neuron models, where the exploration is done by a perturbation of the conductance of the neuron [FS06].

3.4.3.1 Learning in Recurrent Networks

So far, only feed-forward classifiers on more or less complex patterns were discussed. In this section, I want to discuss learning in recurrent networks. This setup if of particular interest, because in biological neuronal networks, at least some degree of feedback input can be expected.

3.4.3.1.1 Hopfield networks

Hopfield networks [Hop82, Hop07] are arguably the most simple setup for a recurrent network: A population of N rate neurons is interconnected by weights w_{ij}, which define the recurrent input. The activation of each neuron i is binary with $S_i \in \{-1, 1\}$, where $S_i = 1$ is an activated state and $S_i = -1$ is a silent state. In this setup, the input into each neuron is given by

$$h_i = \sum_j w_{ij} S_j \tag{3.43}$$

The activation of the neuron will then be defined by an activation function, which is chosen to be the sign function, such that

$$S_i := \operatorname{sgn}\left(\sum_j w_{ij} S_j\right) \tag{3.44}$$

Activations of neurons can either all be updated the same time in a synchronous update or one after the other in an asynchronous update.
Imprinting P patterns ξ^μ onto this network can generate stable patterns, that are self-consistent in that they generate input into each neuron that is compatible with its own state. Patterns can even be completed, if a noisy version of the pattern is presented to the network, because an attractor around the original pattern is formed.
In the Hopfield model, a generalized Hebb rule is employed for learning:

$$w_{ij} = \frac{1}{N} \sum_{\mu=1}^{P} \xi_i^\mu \xi_j^\mu \tag{3.45}$$

With this learning rule, a number of patterns can be stored in such a neuronal network, such that these patterns form attractors. This is a very simple form of content-addressable memory and associative learning.

3.4.3.1.2 Temporal sequences of patterns

So far, stationary patterns were discussed. However, in biological neuronal networks, sequences of activation patterns are essential. In modelling studies, these activation sequences are usually considered in closed loop situations, where the last activation pattern in the sequence restarts the sequence (limit cycles). This process can be seen as a stable trajectory in a phase space, where the vector of the states of the neurons is the state of the system. This state vector then follows a closed trajectory.
Recurrent neuronal networks, which are able to create continuous, self-sustained patterns of activity, are highly sensitive to noise. Hence, any learning algorithm that attempts to teach elongated activity sequences to recurrent networks needs to take this sensitivity into

account.
Laje and Buonomano introduce a model that produces robust patterns in recurrent neural networks in [LB13]. Their model is based on firing-rate units and generates locally stable trajectories in the phase space.
They use innate trajectories through the phase space as a starting point, that is, sequences of activation that the network generates due to the initialization. During a training period, they present the network with noisy input and apply a learning rule similar to the delta rule to achieve an output equivalent to the original trajectory. Thus, they generate areas of attraction around the desired trajectory, from which the network evolves back onto the trajectory. Thus, noise during learning is beneficial for the stability of learned trajectories. To derive meaningful output from this recurrent network, they train an output neuron to respond to the learned pattern in a specific way.
Brea et al. [BSP13] introduce another model based on spike-response model neurons with stochastic spiking. They use a set of visible neurons that are part of the training pattern and a set of hidden neurons that can spike at liberty. Hidden neurons enable networks to solve more complex problems, that are not solvable with visible neurons only. In their study, they introduce a membrane potential

$$u_i(t) = u_0 + \sum_j^N w_{ij} x_j^\epsilon(t) + x_i^\kappa(t) \tag{3.46}$$

where w_{ij} is the synaptic strength from neuron j to neuron i, $x_k^\alpha(t) = \sum_{s=1}^\infty \alpha(s) x_k(t-s)$ represents the convolution of spike train x_k with kernel α and u_0 is the resting potential. The postsynaptic kernel is given by $\epsilon(s) = \frac{1}{\tau_1 - \tau_2}\left(\exp\left(-s/\tau_1\right) - \exp\left(-s/\tau_2\right)\right)$ and the adaptation kernel by $\kappa(s) = c\exp(-s/\tau_r)$ for $s \geq 0$. Both kernels are zero for $s < 0$.
The spiking process is modelled as a stochastic process based on the deterministic membrane potential. The spiking probability of neuron i in time bin t is given by

$$P(x_i(t) = 1 | u_i(t)) = \rho(u_i(t)) \tag{3.47}$$

with $\rho(u) = \frac{1}{1+\exp(-\beta u)}$.
In this setup, they develop a learning rule from the goal that the distribution of output spike trains in the visible neurons $P_w(v)$ is as similar as possible to the target distribution $P^*(v)$. To that end they calculate a learning rule that performs a gradient descent on an upper bound on the Kullback-Leibler divergence given by

$$D(P^*(v)||P_w(v)) = \langle \log \frac{P^*(v)}{P_w(v)} \rangle_{P^*(v)} \tag{3.48}$$

The derived learning rule is given by

$$\Delta w_{ij}^{batch} = \eta \sum_{i=1}^T g_i(t)\left(x_i(t) - \rho_i(t)\right) x_j^\epsilon(t) \begin{cases} 1 & \text{if } i \text{ visible} \\ \log R_w(v|h) - \bar{r} & \text{if } i \text{ hidden} \end{cases} \tag{3.49}$$

where η is the learning rate. $g_i(t) = \frac{\rho_i'(t)}{\rho_i(t)(1-\rho_i(t))}$ with $\rho_i'(t) = \frac{\mathrm{d}\rho(t)}{\mathrm{d}u}|_{u=u_i(t)}$, which implies $g_i(t) = \beta$ for $\rho(U)$ as defined above. $R_w(v|h)$ is the probability of a visible activity pattern, given the past hidden pattern and $\bar{r}$ is a constant. During the learning process, spike trains for the visible neurons are sampled from the target distribution v $P^*(v)$ and imposed on

the visible neurons. Hidden neurons follow the dynamics of the network. The main part of the learning rule is essentially equivalent to the delta learning rule: The difference between the actual and the real activation of each target neuron is correlated with the presynaptic activation.
With this learning rule, a network of spiking neurons can learn to approximate a desired output spike pattern distribution.

3.5 Spike Train Distance Measures

When comparing outputs between neurons, it can be useful to compare the similarity of output spike trains. This can be particularly useful to evaluate learning success in modelling studies.
There are several possible spike train distance measures $d_0(s_1, s_2)$, e.g. the VanRossum-distance [vR01] and the Victor-Purpura-distance [VP96].

3.5.1 VanRossum-Distance

To calculate the VanRossum-distance between two spike trains s_1 and s_2, both spike trains are convoluted with an exponential kernel. Then the quadratic distance is computed between those convolutions. While this spike train distance measure is easy to implement, it has the computational disadvantage of the computing time being dependent on the total number of simulation time steps.

3.5.2 Victor-Purpura-Distance

Calculating the Victor-Purpura-distance seems more complicated, but is generally faster for sufficiently low firing rates: To evaluate a spike train distance between spike trains s_1 and s_2, a cost for the transformation from s_1 into s_2 is calculated. There is a cost of 1 for the deletion or introduction of a spike and a cost of $q\Delta t$ for a shift of the spike time of one spike by Δt, where q is a parameter that scales the cost of shifting a spike relative to the insertion and deletion of spikes. The sum of the costs to transform s_1 into s_2 is then the spike train distance $d(s_1, s_2)$. For an illustration see figure 3.5.1.

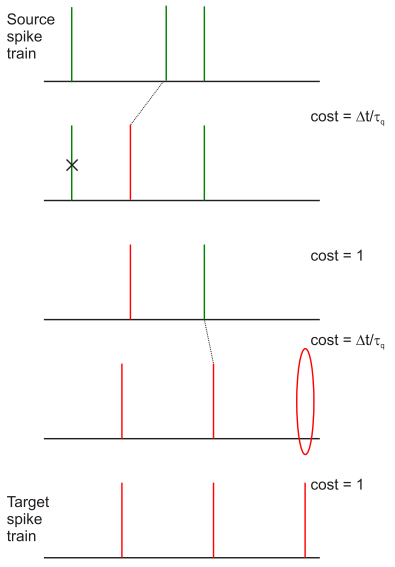


Figure 3.3: Illustration of the Victor-Purpura-Distance

Chapter 4

Theoretical Approaches to Vocal Learning in Songbirds

Imitation learning requires the learning system not only to observe the tutor, but also to have an understanding of its own behaviour. Furthermore, the learning system has to be able to compare the two behaviours and to change its own behaviour accordingly. While imitation learning is a complex process, vocal imitation learning is a relatively well controlled form of imitation learning, in which behaviour is relatively easy to quantify and to compare in a detailed way. Thus, songbirds provide a unique opportunity to study imitation learning experimentally, which can inspire new theoretical work. In fact, vocal imitation learning is one of the simplest forms of imitation learning, because the self-generated action is perceived with the same sense as the tutor action.

While the biological background of songbird imitation learning was discussed in section 2.2, here, I want to give an introduction to the theoretical approaches that have been made towards an understanding of imitation learning in the songbird.

4.1 Reinforcement Learning

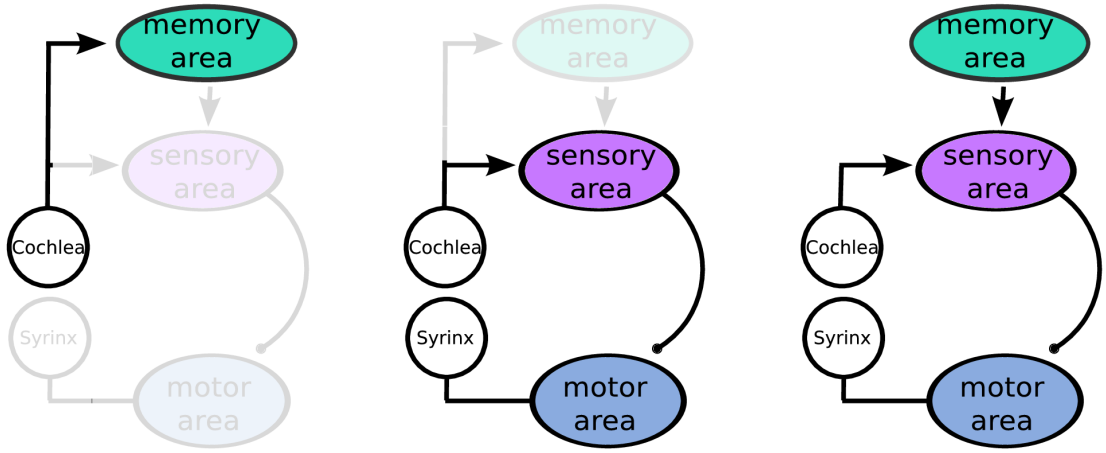
Reinforcement learning is a form of weakly supervised learning. While in supervised learning the network output is controlled by a teacher (see section 3.4.2), in reinforcement learning, an outside entity called the critic only provides feedback on the behaviour of the learning system in form of a reward or a punishment.
The basic idea of reinforcement learning is very simple: trial-and-error learning. The learning system is active in a particular way and then this activity is evaluated by the critic, that gives a reward or punishment. Next, the system tries out a change and if the received reward is higher, it will keep the change and otherwise discard it.
In this way, e.g. irregularly spiking neural networks can estimate the gradient of the reward signal and perform a stochastic gradient ascent on the expected reward [XS04]. Furthermore, reinforcement learning has been shown to be applicable to more realistic neuron models, where the exploration is done by a perturbation of the conductance of the neuron [FS06].

Reinforcement learning has been applied to songbird learning [FFS07] and it was shown that it is useful in learning very simple songs of low dimensionality. This framework

has inspired experimental work that shows that the avian brain area LMAN modulates both, neuronal as well as behavioural, variability during learning, which is essential for the learning process as shown by lesion studies [BMA84, SN91, KDB05, ÖAF05]. Since singing is a form of motor sequence generation, it is tempting to investigate, if there are similarities between the brain structures that allow for acoustic sequence generation in songbirds and brain structures of other birds, unable to sing, but well able to learn behavioural sequences. Pigeons, for example, are able to learn to peck specific sequences on a touch screen. There is a strong neurophysiological similarity between the brain structure of songbirds and of pigeons. Hence, it was tempting to investigate whether the putative pigeon homologue to songbird LMAN provides a similar function, namely the modulation of behavioral variability. However, it was found that pharmacological inactivation of the forebrain area nidopallium intermedium medialis pars laterale (NIML) has no effect on behavioural variability [HWPG14].

4.2 Inverse Models

Inverse models provide a fascinating alternative to reinforcement learning for vocal learning. Since for reinforcement learning, the dimensionality of the activation has to be small and every activation pattern has to be specifically learned, it is unlikely that more complex songs or even human speech is learned this way.
In particular, the remarkable imitation learning skills of particular birds, such as e.g. the lyrebird who is known for imitating chain saws and car alarms, show that reinforcement learning is not or at least not the only way that imitation learning happens in the brain. Also adult humans can perform vocal imitation of sounds in the known realm of sounds that the human can produce - for example by learning to sing a new simple tune. This can usually be done relatively well in one-shot-learning, i.e. immediately and without training.
A possible explanation for these remarkable imitation skills is provided by inverse models: Here, a sensory memory of the tutor song is formed independently from the rest of the learning process. This shifts the problem of learning a motor sequence to learning a sensory sequence, which is simpler to do, because the sensory sequence is input into the learning system as a sequence during the learning process, while the motor sequence needs to be generated without explicit guidance. This process of sequence learning will be discussed in chapter 9.
Independently from the sensory memory formation, in a babbling phase, the young imitator produces random activations in a motor area, which in turn produce some sound, which leads to a sensory activation in a sensory area. During the babbling phase, the imitator learns to associate the sensory impression with the motor activations that caused it by adapting the synaptic connections from the sensory to the motor population. This mapping of sensory activations back onto the causal motor activations is called an inverse model. It can then be used to produce any sound in the realm of the explored space: The sensory memory is just fed through the mapping, such that it will then produce the necessary motor activations (see figure 4.1).
After the inverse model is once formed, any new sound can (to some degree) be imitated quickly, namely as soon as the memory is formed. Hence, one-shot-learning and imitation of very complex sounds becomes possible. In song birds in particular, this means that the tutor song can be reproduced.



(a) During exposure to the tutor song, a sensory memory template is formed in a memory area. A model for this process will be presented in chapter 9.

(b) During a babbling phase, the bird explores its own sound generation capabilities by random exploration and forms an inverse model of its sound generation and perception apparatus. This is the process that will be explored in this chapter, chapter 5 and chapter 8.

(c) After the inverse model is learned, any sensory representation can be reproduced by simply feeding it into the inverse model, which will elicit the corresponding motor activity. This is true in particular for the tutor song.

Figure 4.1: Illustration of the general setup of bird song learning with inverse models.

4.2.1 Existing Learning Rules

Inverse models were suggested as a possible mechanism for vocal imitation learning in song birds by Hanuschkin et al.[HGH14]. In their study, they suggest a simple learning rule that serves to learn an inverse model in a simple linear framework: the sensory response $a(t)$ in the sensory brain area at time t is a vector of firing rates linearly related to the motor cause $m(t-\tau)$ at an earlier time $t-\tau$. m is a vector of firing rates in the motor area, which they assume to be either HVC or LMAN. They assume a constant loop delay $\tau = \tau_m + \tau_a$, where τ_m is the time needed for the motor activation m to generate a sound and τ_a is the time needed for the thus generated sound to elicit a response in a. They go on to assume a linear mapping from motor activity to auditory activity:

$$a(t) = \mathbf{Q}\mathbf{m}(\mathbf{t}-\tau) \tag{4.1}$$

where $\mathbf{Q}$ is a matrix. A successful inverse model V can be described as $V = Q^{-1}$. This mapping then allows to postdict the motor causes m^a of any given sensory target a according to

$$m^a = \mathbf{V}\mathbf{a} \tag{4.2}$$

Thus, this inverse model can be used in feed-forward motor control to generate a motor sequence $m^a(t)$ which in turn generates sound that elicits the desired sensory response $a(t)$. In their study, Hanuschkin et al. suggest a local learning rule which is based on an eligibility trace $e(s)$, which establishes a link between the activity in a motor neuron

at time t and activity in a sensory neuron at a later time $t + s$. This eligibility trace is assumed to decay monotonically over time, which implies that sensory neurons connect preferentially onto motor neurons that were recently and reliably activated rather than onto motor neurons that were activated a long time ago. The decay of $e(s)$ must be slow enough to account for the loop delay τ, such that $e(\tau) >> 0$. Then the learning rule describing changes in synaptic strength V_{ij} from auditory neuron j onto motor neuron i is given by

$$\delta V_{ij} = \int_0^\infty \mathrm{d}s \, [e(s) m_i(t-s) a_j(t)] - \hat{m}_i(t) a_j(t) \tag{4.3}$$

where $\hat{m}(t) = \sum_k V_{ik} a_k(t)$ is the silently postdicted motor activity corresponding to the summed auditory input to neuron i at time t. They describe this silently postdicted motor activity $\hat{m}_i a_j$ as a heterosynaptic depression term; they also assume that this signal is transferred silently, that is without a contribution to synaptic input into motor neuron i. This requires a gating off of sensory input into the motor population during internally generated motor explorations. With this world model and learning rule, they derive that the type of inverse model formed and the type of mirror neuron it gives rise to depends on the type of motor code used for exploration. A causal inverse model can be learned when the motor explorations are random noise, which will in turn give rise to mirror neurons with a time delay between motor and sensory activation which is equivalent to the loop delay τ. This type of causal inverse model maps the auditory activation $a(t)$ back onto its cause $m^a(t)$, such that with this inverse model any sound can be imitated by feeding its corresponding activation pattern $a(t)$ into the inverse model.

A predictive inverse model can be learned by stereotyped neural input during learning, which gives rise to mirror neurons with a time delay of zero. Since predictive inverse models just map the auditory activation $a(t)$ at time t onto a motor activation $m(t + \delta t)$ at a slightly later time $t + \delta t$, which repeatedly occurs during the stereotyped neural activations used during learning, auditory activations are not mapped onto their own motor causes, but onto the motor causes of subsequent auditory activations, which limits the set of imitable sounds to the training sequence. Hence, predictive inverse models are not useful for imitation of arbitrary sounds.

While this study provides a local learning rule which allows first insights into the learning of inverse models and their connection to mirror neurons, it remains unclear how the silent signal for the heterosynaptic plasticity $\hat{m}$ could be propagated. Also, it is unclear, how the difference between this silent signal $\hat{m}$ and the actual motor activation weighted by the eligibility trace is computed. Furthermore, it would be interesting to extend the learning of inverse models to spike-based neural codes as opposed to firing rate based neural codes.

4.2.2 Inverse Models in Spiking Neurons as a Form of Pattern Association

If we consider inverse models in spiking neurons, any model of the sound generation and perception process will be non-linear due to the threshold property of the spiking neurons. In the most general sense the sound generation and perception process can be described as mapping from a spike pattern in the motor population to a spike pattern in the sensory population: A particular pattern of motor activity will induce a certain movement in the respiratory muscles and the syrinx, which will in turn generate a specific sound which will

be perceived by the auditory system in a certain way to generate a matching sensory spike pattern. Due to the threshold nature of spiking neurons, this mapping is always non-linear. Furthermore, the mapping may or may not be local in time: Both, the generated sound and the sensory activations might depend on events in the past.
An inverse model in this context will then be the mapping from the self-generated sensory pattern back onto the motor pattern that generated it. In the case of a causal inverse model, each particular part of the sensory pattern will be mapped back onto the motor pattern that produced it.
In this framework, the learning of such an inverse model thus shares a lot of properties with the problems of pattern association. In fact, it is just a particular case of pattern association on a very specific set of patterns.
While in pattern classification problems in feed-forward networks mostly the entire set of patterns is presented during learning (see e.g. [HKP91]), this is impossible to do in the inverse model, because of the large number of possible motor patterns. The inverse model thus has to generalize between the patterns used during learning to be able to invert the entire pattern space.
The learning of inverse models of forward mappings which are local in time, i.e. where the sensory activation only depends on the motor activation at a given time, is very similar to Perceptron learning: if the motor activation and the sensory activation is binned into (arbitrarily small) time bins, for each moment in time a motor pattern evokes a particular sensory pattern. This pattern then has to be matched back onto the original motor pattern. To that end, each motor neuron has to learn to respond to a particular spatial pattern of sensory activation either by spiking or by not spiking (see figure 4.2(a)). While the set of patterns may be very large, this is just the Perceptron problem.
If the forward mapping is not local in time, each motor spike influences or elicits not only a local response, but a elongated pattern of sensory activity. This elongated pattern then needs to be mapped back onto a spike in the relevant motor neuron at a precisely defined time and no spikes in any other neuron. This is a setting very similar to the Chronotron problem, albeit with a very peculiar set of patterns (see figure 4.2(b)).
Hence, the learning of inverse models can be investigated in the framework of pattern classification.

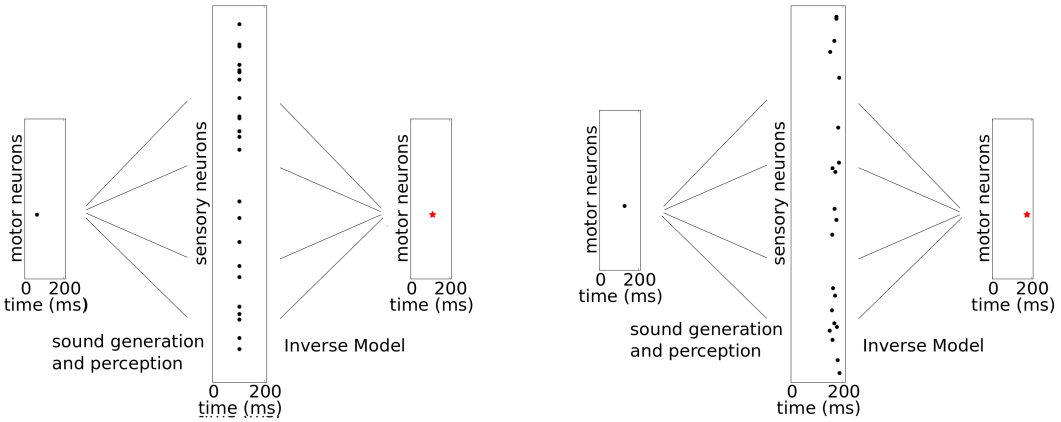


Figure 4.2: The learning of inverse models in spiking neurons can be mapped onto the learning of pattern association. In each of the figures, left is a raster plot of the simplest possible motor activity, which is then transformed by the sound generation and perception process to a pattern of sensory activation, which is displayed in the middle raster plot. Finally, a successful inverse model maps this sensory pattern back onto the original motor activation, possibly with a time shift due to the delays between motor and sensory populations (right raster plot, red star). (a) For models of the sound generation and perception that are local in time, the learning of an inverse model is similar to the Perceptron problem (see chapter 5 and chapter 6). (b) For models that are non-local in time, the learning of the inverse model is similar to the Chronotron problem (see chapter 7 and chapter 8.)

Chapter 5

Learning of Inverse Models with Anti-Hebbian Spike Timing Dependent Plasticity

5.1 Introduction

Inverse sensor-motor models serve to generate a desired sensory input by appropriate motor actions. In this sense, they attempt to "invert" the action-sensation mapping given by the physical world. While in general this mapping is not stationary, sound sequence imitation represents a comparatively well controlled situation. Therefore, it was tempting to propose inverse models as the mechanism enabling many bird species to imitate previously heard acoustic signals [HGH14]. The underlying hypothesis is that inverse models in the bird's brain perform a transformation of memorized sensory representations of sound sequences into spatio-temporal patterns of activities in motor areas that in turn generate the same sound sequences. This enables imitation of arbitrary sound sequences within the realm of the possible sounds the bird can produce. A crucial prediction of such so called causal inverse models are mirror neurons active during both singing as well as playback of a recording of the bird's song. The responses of these mirror neurons to a playback would be delayed relative to the bird itself singing the song. This delay reflects the loop time it takes for motor activations to produce sound, which produces sensory activations that are looped back to the respective motor area. Indeed, a recent study has found evidence for such delayed mirroring in area LMAN of the song bird [GKGH14] (for more information on auditory-vocal mirroring in songbirds see 2.2).

The loop time varies between songbird species in a range from a minimum of around $35ms$ in the LMAN loop in zebra finches [GKGH14] to the order of magnitude of $70-90ms$ measured in behavioural studies with altered feedback in Bengalese finches [SB06]. In this thesis, a loop delay of about $42ms$ is chosen for the model to match the time delay found in the mirror neurons in LMAN in zebra finches[GKGH14].

The classical form of spike-timing-dependent plasticity (STDP) for excitatory synapses (here denoted CSTDP) postulates that the causal temporal order of first presynaptic activity and then postsynaptic activity would lead to long-term potentiation of the synapse (LTP) while the reverse order would lead to long-term depression (LTD)[DP04, DP06,

CD08]. More recently, however, it became clear that STDP can exhibit different dependencies on the temporal order of spikes. In particular, it was found that the reversed temporal order (first post, then presynaptic spiking) could lead to LTP (and vice versa; RSTDP), depending on the location on the dendrite [FPD05, SH06]. For inhibitory synapses some experiments were performed which indicate that also here STDP exists and has the form of CSTDP [HNA06]. Note that CSTDP in inhibitory synapses in its effect on the postsynaptic neuron is equivalent to RSTDP of excitatory synapses. For simplicity, here the presentation is restricted to RSTDP for synapses that in contradiction to Dale's law can change their sign.

Under natural conditions synaptic changes caused by STDP will depend not only on the inputs but also on the dynamical properties of the pre- and postsynaptic neurons within a network. One example for a beneficial effect of such an interplay was investigated in [DLH10], where CSTDP interacted with spike-frequency adaptation of the postsynaptic neuron to perform a gradient descent on a square error. Several other studies investigate the effect of STDP on network function, however mostly with a focus on stability issues (e.g. [SMA00, ID03, VSZ$^+$11]). In contrast, here the focus is put on the constructive role of STDP for the learning of inverse models.

5.2 The Model

To investigate learning of inverse models, a model of connected neuronal populations reflecting the brain anatomy of songbirds is constructed (see figure 5.1). A neuron population m in the motor area activates the muscles in the syrinx for singing. The bird's cochlea converts sounds into activations of neurons in sensory area s. To model the bird hearing its own vocalizations, spatio-temporal activity in m is converted to input in s through one or several delayed linear transformations. This gives self-generated input into the sensory area at or around the loop delay. Mimicking the "babbling" young birds presumably use to establish the relation of motor activities with the corresponding sensations of self-generated sounds, the motor neurons are driven with noise during an exploration phase (see figure 5.3(a) for an example pattern). Consequently, the spatio-temporal motor activity is transformed into input into the sensory neurons, which in turn create spatio-temporal sensory spike patterns (see figure 5.3(b)).

A successful inverse model then has to map these sensory patterns, when retrieved from memory, back onto the motor patterns that have generated the respective sensory inputs.

There are several fundamental challenges when it comes to learning a causal inverse model:

- During exploration, there must be some trace of exploratory motor activation over time to account for the loop delay.
- This trace of the past motor activation must be compared to the input generated by the inverse model.
- It would be interesting to study this in a framework of spike-time-based neural code rather than in a rate-based neural code.
- All ingredients to the model should be biologically plausible; in particular the learning rule should be local.

This model attempts to achieve all of this.

5.2.1 Neuron Model

Neurons are modelled as simple leaky integrate-and-fire neurons in both areas m and s. Area m contains N_m neurons, which will be indexed by j, area s consists of N_s neurons indexed by i. Their membrane potentials V (which are all considered to be measured relative to the resting potential set to zero) are governed by

$$\tau_m^m \dot{V}_j(t) = -V_j(t) + I_j(t) + I_{explor} \tag{5.1}$$

and

$$\tau_m^s \dot{V}_i(t) = -V_i(t) + I_i(t) + I_{noise} \tag{5.2}$$

respectively, where $\tau_m^{m/s}$ is the membrane time constant and $I_{i,j}$ is the external current generated by the network, I_{explor} is the current used to elicit activity during exploration and I_{noise} is a current induced by potential background noise, which will be set to zero unless otherwise mentioned. In the absence of input currents, the membrane potential decays back to the resting potential at $V_{rest} = 0$. I_j consists of synaptic inputs from s, weighted by their synaptic strength w_{ji}, such that each presynaptic spike induces a synaptic current that then leads to an increase or decrease of the membrane potential with a time delay of $\tau_{sm} = 2ms$:

$$\tau_{syn} I_j = -I_j + \sum_i w_{ji} s_i(t - \tau_{sm}) \tag{5.3}$$

I_{explor} for each motor neuron is independently generated: poissonian spikes are generated with an exploration firing rate r_{explor} and then fed into I_j as regular synaptic inputs with weight 1. Every time a membrane potential crosses the firing threshold $V_{thresh} = 2mV$, a spike in this neuron is registered and the voltage is reset to the reset potential $V_{reset} = -48mV$. Spike trains are written as a sum of delta pulses

$$m_j(t) = \sum_k \delta(t - t_k^j) \tag{5.4}$$

and

$$s_i(t) = \sum_k \delta(t - t_k^i) \tag{5.5}$$

where t_k^i is the time of the k-th spike in neuron i. To provide the hyperpolarisation essential to this model, the membrane time constant $\tau_m = 50ms$ is chosen to be relatively long as to provide a substantial remaining hyperpolarisation at the time of the self-generated input from s. This is also supported by a low reset potential.

5.2.2 World Model

To model the bird hearing its own vocalizations, spatio-temporal activity in m is converted to input in s through one or several delayed linear transformations, where N_w is the number of different delays and τ_w is the temporal difference of the delays relative to each other. To construct this model of the world, a sparse matrix M_{all} is created, where each entry

is either zero, a positive constant with probability $P_p = 0.1$ or a negative constant with probability $P_n = 0.1$. Then N_w empty matrices are constructed and the content of M_{all} is distributed over these matrices with equal probability by assigning each entry of M_{all} a delay from the set of the N_w different delays associated with the N_w matrices M_r, such that finally $\sum_r M_r = M_{all}$. This construction of the world ensures maximum comparability between the most simple case of just one delay and the more complex ones.

To then generate input into the sensory population s, spikes in m are low-pass filtered by $\tau_s \dot{\vec{y}} = -\vec{y} + \vec{m}$ and used as input into the sensory population.

$$I_i(t) = \sum_r^{N_w} M_r^i \vec{y}(t - \tau_{ms} - (r - \frac{1}{2}N_w)\tau_w). \tag{5.6}$$

This generates a nonlinear transformation of motor activities into sound activities, which may or may not be local in time.

5.2.3 Spike-Timing Dependent Plastictiy

Synapses from sensory area s to motor area m are denoted by their synaptic weights w_{ij}. They are plastic according to reverse spike-timing dependent plasticity (RSTDP) and of a delay of $\tau_{sm} = 2ms$. Spikes in the sensory area are denoted as $\vec{s}(t)$ and spikes in the motor area m as $\vec{m}(t)$, both of which consist of a sum of delta pulses, see equations 5.4 and 5.5. Additionally, a spike-like event in the motor neurons is introduced purely for plasticity purposes: on the crossing of an additional plasticity threshold $0 < V_{thresh}^{plast} < V_{thresh}$ from below, a spike-like event is registered and will be denoted $\vec{m}^{st}(t)$. When a spike arrives at the synapse with weight w_{ij}, it leaves a trace $\bar{m}_i(t)$ resp. $\bar{s}_j(t)$:

$$\begin{aligned} \tau_{pre}\dot{\bar{s}_j} &= -\bar{s_j}(t) + s_j(t) \\ \tau_{post}\dot{\bar{m}_i} &= -\bar{m_i}(t) + m_i(t) \ . \end{aligned} \tag{5.7}$$

RSTDP can then be modelled by a suitable interaction of spikes and traces:

$$\dot{w}_{ij} \propto -\bar{s_j}(t) m_i^{st}(t) + \bar{m_i}(t) s_j(t) \ , \tag{5.8}$$

where $\tau_{pre} = 4ms$ and $\tau_{post} = 50ms$ define the time course of the STDP window. Probing this plasticity rule with spike pairs reveals that it reproduces the anti-symmetric STDP window (see figure 6.1). Because the differential equations (6.8) are linear, the resulting STDP rule is also linear. All weight changes are effective immediately and weighted with a learning rate η. To allow for maximum comparability between system sizes, the learning rate is scaled with the system size, such that $\eta_{norm} = N_s\eta = 0.00025$ is constant for all trials unless otherwise mentioned.

5.2.4 Measuring the Learning Progress

5.2.4.1 General Measuring Procedure

To evaluate learning progress, one particular motor pattern with input rate r_{song} into the motor area during exploration and its respective sensory pattern are picked out and assigned the role of the tutor song; they are stored for later comparison. This is done to ensure that the model bird can in principle generate the desired activity sequence and

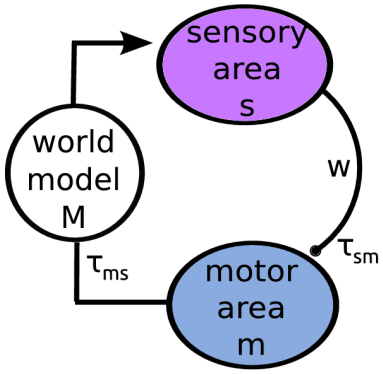


Figure 5.1: Sketch of the system setup. A motor population m feeds into a sensory population s via a world model M, which models the sound generation and perception process with one or several weighted delay lines with or centered around the delay $\tau_{ms} = 40ms$. The sensory population s feeds back into the motor population m via the inverse model w with a delay $\tau_{sm} = 2ms$.

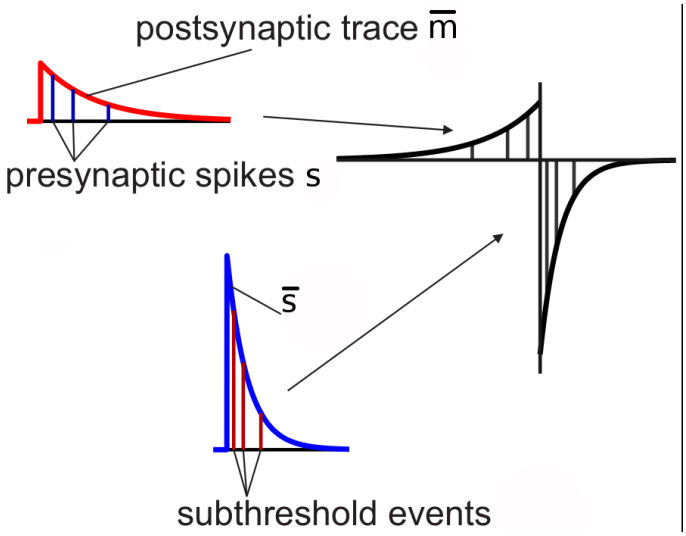


Figure 5.2: Learning rule (5.8) is equivalent with RSTDP. A postsynaptic spike leads to an instantaneous jump in the trace $\bar{m}$ (above left, red line), which decays exponentially. Subsequent presynaptic spikes (dark blue bars and corresponding thin gray bars in the STDP window) "read" out the state of the trace for the respective $\Delta t = t_{pre} - t_{post}$. Similarly, postsynaptic spikes m^{st} read out the presynaptic trace $\bar{s}$ (lower left, blue line). Sampling for all possible times reconstructs and assembles the traces in the STDP window.

is equivalent to assuming the tutor bird and the student bird to have the same mapping from motor activity to sound output. The firing rate of exploration during learning r_{explor} can be the same or different from the song firing rate r_{song}. Learning and recall periods are of duration $T = 3000ms$. Weights are initialized as zero, the world model initialized at random for every trial according to section 5.2.2.

To allow for learning, in the exploration phase, random input is given into the motor population m during $N_k = 5000$ learning epochs. During learning, all weight changes are immediately applied. Every $\Delta N_k = 100$ learning epochs, recall is tested.

To test the ability of the system to reproduce the tutor song, the sensory representation of the tutor song is set to be the activity in the sensory area s by hand (in addition to any self-generated activity). This sensory activity then generates some motor activity in area m, which - if learning was successful - should be a shifted version of the tutor motor activity. This recall motor activity is in turn fed into a copy of the sensory population to test whether it generates the same sensory impression on the model bird that the tutor

song did. Testing whether the same sensory impression is generated is equivalent to testing if the song sounds the same to the (model) bird, which is the marker of good imitation. It is in principle possible that two very different motor sequences generate the same sound output and therefore the same sensory impression. Since for the bird the emphasis is on mimicking the sound, this is the relevant measure of success. As we will see, however, the difference between learning success measured on the motor patterns and learning success measured on the sensory patterns is very small, if learning is successful.

5.2.4.2 Measure of Pattern Similarity

To quantify the similarity of the tutor pattern and the self-generated pattern, it is necessary to compare two sets of spike trains. The activity $a_i^s(t)$ in each neuron i in the tutor song will have to be compared to the activity during recall $a_i^r(t)$ to give some distance measure $d_0(a_i^s(t), a_i^r(t))$. The total distance over the activity a^s resp. a^r of all neurons in the given population will then just be the sum over all neurons in the population. Finally, this distance should be minimized over a global shift to account for the loop delay:

$$d(a^s, a^r) = \min_{\Delta t} \sum_i d_0(a_i^s(t), a_i^r(t - \Delta t)) \tag{5.9}$$

This quantity is evaluated every $\Delta N_k = 100$ learning cycles. The resulting learning curves are normalized to the number of spikes in the tutor pattern, such that the error before learning is 1. For small N_s, it is possible that the sensory tutor pattern does not contain any spikes. Since this is equivalent to a tutor song without sound, these trials are discarded and repeated with a different initialization.
For the quantitative analysis, the residual error after learning is computed by taking the average over the last 10% of learning steps in each of the $N = 50$ trials and then computing average and standard error from those measurements.

5.2.4.3 Spike Train Distance Measures

There are several possible spike train distance measures $d_0(s_1, s_2)$, e.g. the VanRossum-distance [vR01] and the Victor-Purpura-distance [VP96].

To calculate the VanRossum-distance between two spike trains s_1 and s_2, both spike trains are convoluted with an exponential kernel. Then the quadratic distance is computed between those convolutions. While this spike train distance measure is easy to implement, it has the computational disadvantage of the computing time being dependent on the total number of simulation time steps.

Calculating the Victor-Purpura-distance seems more complicated, but is generally faster for not too high firing rates: To evaluate a spike train distance between spike trains s_1 and s_2, a cost for the transformation from s_1 into s_2 is calculated. There is a cost of 1 for the deletion or introduction of a spike and a cost of $q\Delta t$ for a shift of the spike time of one spike by Δt, where q is a parameter that scales the cost of shifting a spike relative to the insertion and deletion of spikes. The sum of the costs to transform s_1 into s_2 is then the spike train distance $d(s_1, s_2)$.

5.2.5 Autocorrelation Function

In the setup of the model, the membrane time constant of neurons in the motor population is very long, which leads to a imposed distance between spikes in these neurons. To be able to quantify this experimentally accessible property of the model, the autocorrelation function is introduced.

To evaluate how the spiking probability of the motor neurons depends on past spiking activity, consider the autocorrelation function $\rho(\Delta t)$:

$$\rho(\Delta t) = \frac{\langle (m(t) - \bar{m})(m(t - \Delta t) - \bar{m}) \rangle_{N_m N}}{\sigma_m^2} \tag{5.10}$$

where $< \cdots >_{N_m N}$ denotes the average over motor neurons and the ensemble, $\bar{m}$ is the mean and σ_m is the standard deviation of $\vec{m}(t)$. In this form, the autocorrelation function is normalized between -1 and 1, 1 indicating perfect correlation, -1 indicating perfect anti-correlation. An autocorrelation of 0 indicates uncorrelated spiking activity.

5.3 Results

To investigate learning of inverse models, a model of connected neuronal populations reflecting the brain anatomy of songbirds is constructed (see figure 5.1). A population m in the motor area activates the muscles in the syrinx for singing. The bird's cochlea converts sounds into activations of neurons in sensory area s . To model the bird hearing its own vocalizations, spatio-temporal activity in m is converted to input in s through one or several delayed linear transformations. This gives self-generated sounds at or around the loop delay. Mimicking the "babbling" phase, the motor neurons are driven with noise during an exploration phase (see figure 5.3(a) for an example pattern). Consequently, the spatio-temporal motor activity is transformed into input into the sensory neurons, which in turn create spatio-temporal sensory spike patterns (see figure 5.3(b)).

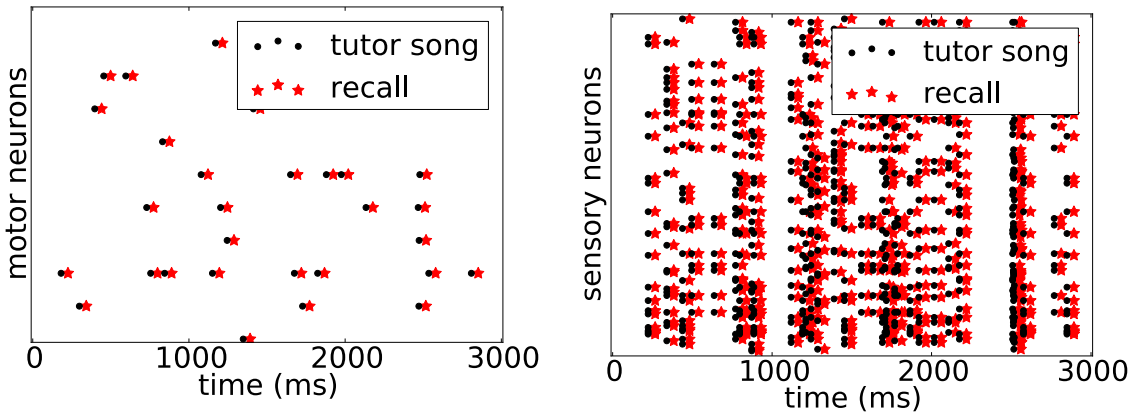
A successful inverse model then has to map these sensory patterns back onto the motor patterns that have generated the respective sensory inputs.

5.3.1 Intuitive Understanding of the Learning Process

To facilitate an intuitive understanding of the learning process, I will here first present the learning principle.

Consider just one spike in one motor neuron at a particular time. This spike will lead to some sort of activation in the sensory population with a delay of τ_{ms}. With a delay of τ_{sm}, this activation will provide input back into the motor neuron.

Before learning, due to hyperpolarisation after each spike in a motor neuron its membrane potential is well below resting potential at the time of the recurrent input from the sensory population. Furthermore, for the respective synapse, first a postsynaptic spike happens and then a presynaptic one in each sensory neuron that was activated due to the motor activity. This order of spiking leads to a potentiation of the synapse, such that upon repetition of the same activation patterns the membrane potential will be higher. This process will continue, until the membrane potential hits the plasticity threshold V_{thresh}^{plast}. Then, the neuron will perceive a postsynaptic spike, a presynaptic spike and a postsynaptic plasticity event. This will in net effect cause a weakening of the synapse, because the time



(a) motor population (b) sensory population

Figure 5.3: Raster plots of an example of activations in the motor (left) resp. sensory population (right) for $N_m = 10$, $N_s = 200$, $N_w = 1$ and $r_{explor} = r_{song} = 1Hz$. Black dots represent spikes in the tutor song, red stars represent recall spikes. Recall spikes are shifted for about τ_{loop} with respect to tutor spikes.

distance between the postsynaptic spike and the presynaptic spike is much larger than the time distance between the presynaptic spike and the postsynaptic plasticity event and because of the time constants of the plasticity traces. This will halt plasticity such that the membrane potential will stay below the threshold at that point in time. However, there will be a substantial amount of extra input back into the motor neuron from the sensory population at $\tau_{loop} = \tau_{sm} + \tau_{ms}$ after the original motor spike.
During recall, the original motor spike is absent, such that the extra input from the sensory population drives a motor spike in the right neuron with a time shift of τ_{loop}.
For all other motor neurons, no spike happens. All input from the sensory population will be suppressed as soon as the membrane potential reaches the plasticity threshold.
For more complex patterns, the same process applies: Whenever a motor spike occurs, the synapses from the corresponding sensory neurons are potentiated, until the membrane potential of the motor neuron reaches the plasticity threshold, whereupon plasticity is stopped. Figure 7.3 shows the membrane potential in five motor neurons during learning (left column) and during recall (middle column), as well as the complete song pattern in all $N_m = 10$ motor neurons (right column, black dots) and the recall of that song pattern (right column, red stars) over the course of learning. During the learning process, in the learning trials, the membrane potential shows a more and more pronounced upwards deflection of the membrane potential at τ_{loop} after each spike. During the recall trials, first only very small, sub-threshold upwards deflections of the membrane potential become visible that grow as learning continues, until they reach the threshold and cause a recall spike. After learning, the entire pattern is completed.

5.3.2 Quantitative Evaluation of the Learning Process

For the quantitative evaluation of the learning process, the spike train distance between patterns of motor resp. sensory activity is measured every $\Delta N_k = 100$ learning epochs over a total of $N_k = 5000$ learning epochs. Learning curves are normalized, such that

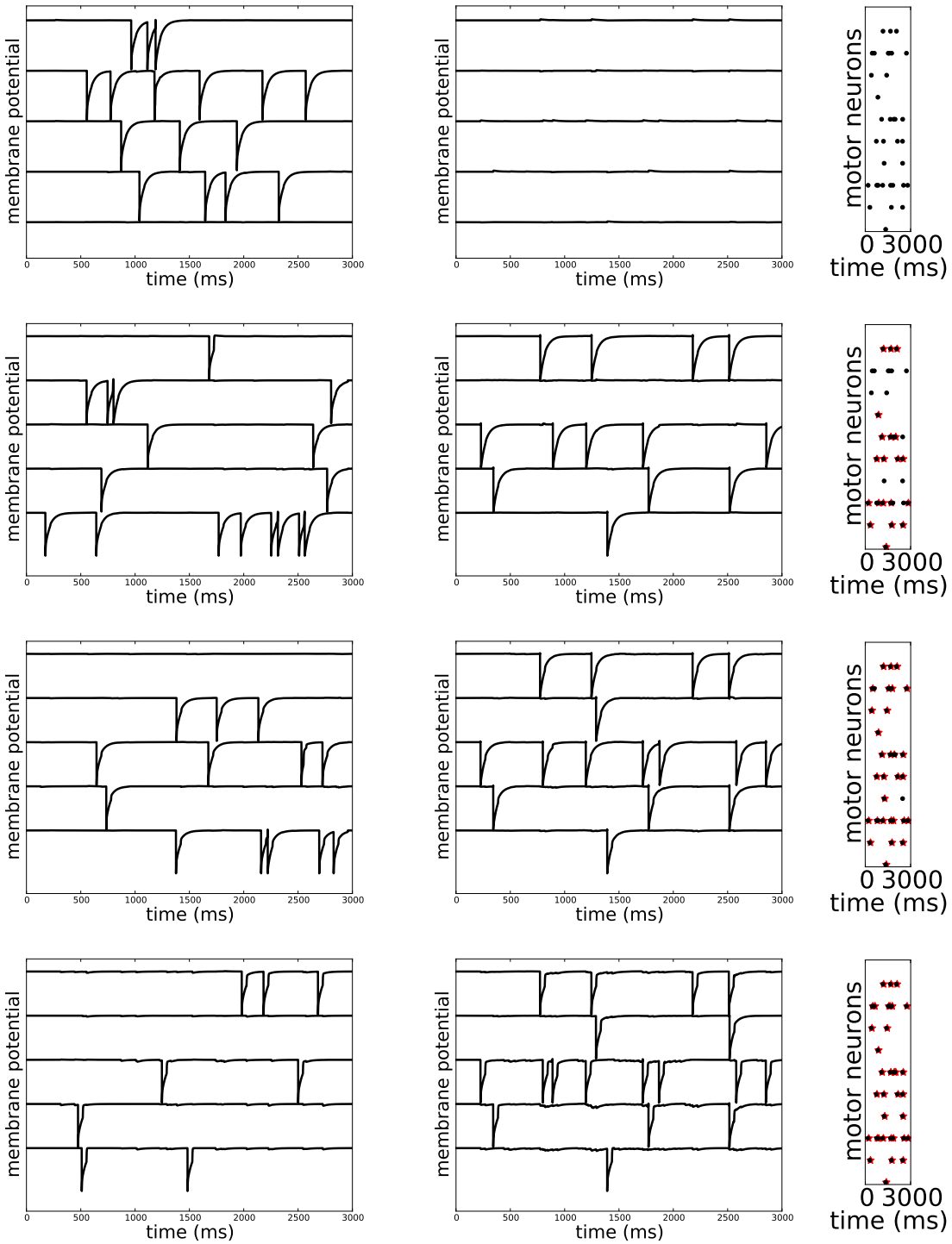


Figure 5.4: From left to right: Example of membrane potential during training, example of membrane potential during testing, spike raster plot of pattern in m (black: tutor pattern, red: recall pattern). Top to bottom: At learning steps 10, 20, 50, 200. During learning, the hyperpolarisation is filled up more and more by extra input from the sensory population. In the recall case, the teacher spikes are missing, such that the extra input drives the motor neurons over the threshold; the inverse model is learned.

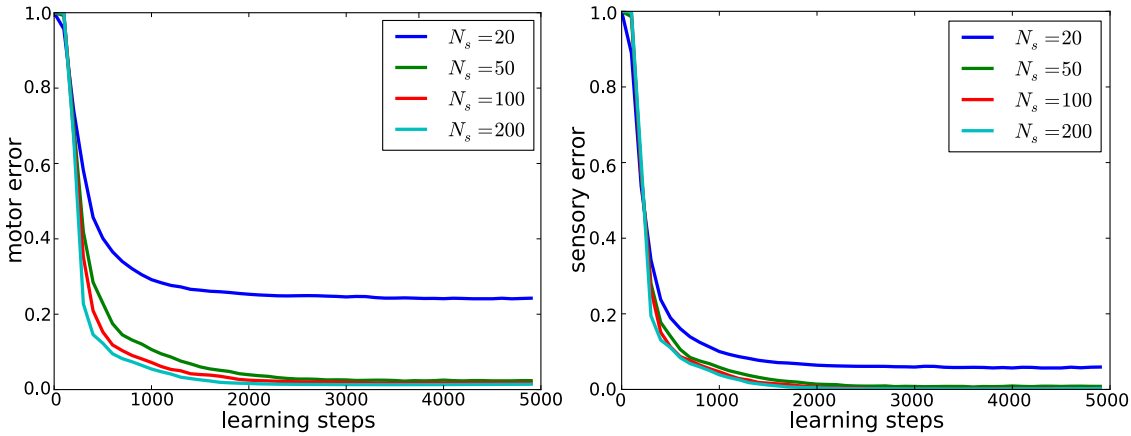


Figure 5.5: Learning curves for $N_m = 10$ and different sensory population sizes. Over learning, the error decreases quickly and settles on a low level. For large errors after learning, the error is more pronounced in the motor population than in the sensory population. The residual error is calculated by taking the average over the last 10% of learning steps for individual learning curves. From these residual errors average and standard error are calculated. Left: motor population, right: sensory population.

the error before learning is 1. This is equivalent to a normalization to the number of spikes in the tutor pattern. A typical set of learning curves, averaged over $N = 50$ sets of initializations, for an exploration and song firing rate $r_{explor} = r_{test} = 1Hz$ is displayed in figure 5.5. Learning is quick and after learning, the system settles at a low error. When learning is successful, learning curves for the motor pattern and the sensory pattern are similar. Note that in case of a relatively high residual error ($N_s = 20$, blue line), the error is higher in the motor population than in the sensory population. This is a trace of the fact that different motor patterns do not give sufficiently different sensory patterns, such that the mapping is difficult to invert.
For all further evaluations, a residual error is calculated by taking the average of the last 10% of learning steps for each initialization. From this set of residual error measurements, average and standard error are computed and used in all further investigations.

5.3.3 Dependency on System Size

To explore how the residual error after learning depends on the system size, several different sizes are tested with $N_m = \{10, 15, 20, 25, 30\}$ for an exploration and song firing rate $r_{explor} = r_{test} = 1Hz$ and the simple world model with $N_w = 1$. Since it is reasonable to assume that the residual error will scale with the ratio $\alpha = N_s/N_m$ of neuron numbers in s and m, residual error after learning is computed for $\alpha = \{0.2, 0.4, 0.6, 0.8, 1, 2, 5, 10, 20\}$. The results for different system sizes are displayed in figure 5.6. Learning is increasingly successful for increasing α and settles at about $\alpha_c = 10$. Learning success depends only very little on system size with lower residual error at low α for larger systems.

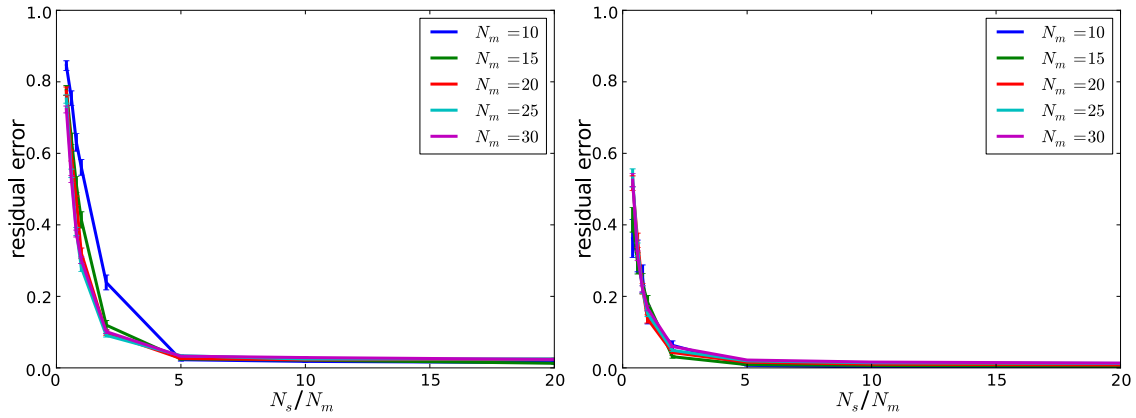


Figure 5.6: Residual error after learning in dependency on size ratio between sensory and motor populations. Left: motor population, Right: sensory population.

5.3.4 Dependency on Firing Rates

In the songbird, in most areas in the song system firing rates are higher than $1Hz$, so it is important to investigate learning also for higher firing rates. Due to the long after-hyperpolarisation in the motor area, spiking is prevented immediately after each spike for at least the time of the loop delay $\tau_{loop} = \tau_{ms} + \tau_{sm} = 42ms$. Since the length of the hyperpolarisation in the motor population is chosen at $\tau_m = 50ms$ to provide a reasonably strong distance of the membrane potential to the threshold at the time of the self-generated input, the maximum firing rate is limited to at most $20Hz$, if the spiking is entirely regular. Here, the investigation of learning success is limited to the range of firing rates of $r_{explor} = r_{song} = \{1Hz, 2Hz, 3Hz, 4Hz, 5Hz, 6Hz, 7Hz\}$. To investigate how the success of learning depends on the motor firing rate, the residual error is calculated for different firing rates for $N_m = 10$ and $\alpha = \{2, 5, 10, 20\}$ and the simple world model with $N_w = 1$. The results are displayed in figure 5.7. The residual error is low for low firing rates up to $3Hz$ and then rises to moderate levels with the firing rate. Note, however, that the quantitative results differ from motor to sensory population with the overall learning success higher in the sensory population. So even though there may be differences in how a sensory impression was generated, the sensory impression is very similar.

5.3.5 Learning with Background Noise

Since songbirds usually learn to sing in large and noisy bird colonies, it is interesting to investigate, how well inverse models can be learned with background noise. To that end, extra random input is fed into the sensory population. This input mimics extra sensory input by being of the same type as the input that reaches the sensory neurons via the model of the world. The rate with which this input is given is calculated in percent based on an estimation of the firing rate during song r_s^{est}: The number of spikes in the song are divided by the duration and the size of the sensory population:

$$r_s^{est} = \frac{1}{N_s T} \sum_j \int_0^T s_j(t) \mathrm{d}t \tag{5.11}$$

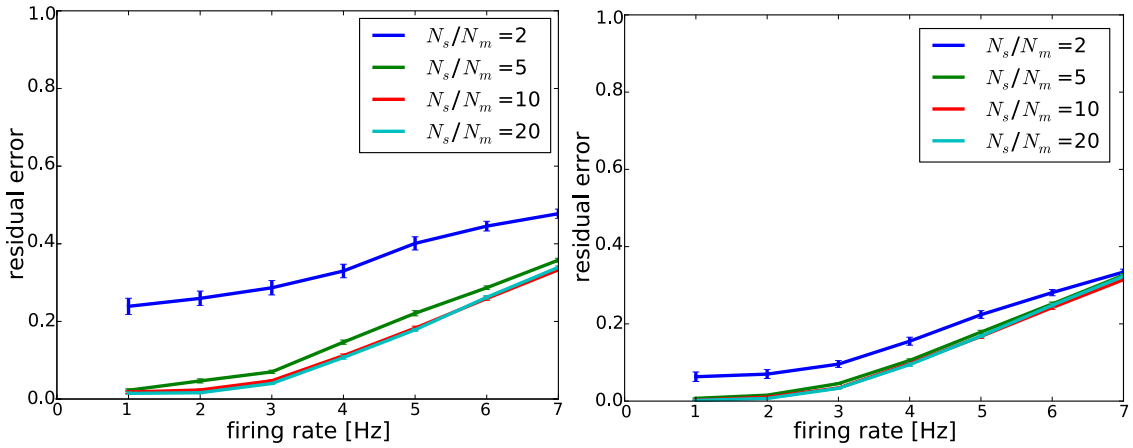


Figure 5.7: Residual error for different size ratios between sensory and motor populations in dependency of exploration firing rate. Song firing rate is equal to exploration firing rate. Left: motor population, Right: sensory population. The residual error is low for low firing rates and then rises to moderate levels with higher firing rates.

Figure 5.8 shows residual errors for $N_m = 10$ and $N_w = 1$ for different population size ratios $\alpha = \{2, 5, 10, 20\}$ in dependency on the noise level.
While for low α, the influence of noise is substantial, the residual error saturates for noise levels up to 200%. For high α, however, the noise level only has a very low impact on the residual error.

5.3.6 Necessity of Exploration with Testing Firing Rate

Up to this point, it was assumed that the firing rate in the motor population m was the same for both, the exploration phase and the song. Since this would require prior knowledge of the firing rate of the song, it is interesting to investigate, how the residual error changes when exploration is done with a firing rate different from the song firing rate. To this end, the ratio $\beta = r_{song}/r_{expl}$ is introduced. The residual error is measured for $N_m = 10$ and $N_s = 100$ for exploration firing rate $r_{expl} = \{1Hz, 2Hz, 3Hz\}$ and $\beta = \{0.25, 0.5, 1, 2, 4\}$. The results are displayed in figure 5.9.
The residual error is low for low song firing rates and rises for higher song firing rates. Learning is successful as long as the exploration firing rate is higher than the song firing rate, which is consistent with high activation for random exploration. However, the possible firing rates for the motor population remain limited due to the hyperpolarisation.

5.3.7 Towards more complex Models of the World

Up until now, all results were obtained for a very simple model of the world: a simple, weighted delay line. However, it can be assumed that the sound generation and perception process of the songbird is a lot more complex. As a first step towards more complex models of the world, here, several delays from motor to sensory population will be considered. This is done by spreading the input of one delay line from the simple world model over several delays, apart by $\tau_w = 1ms$. For maximal comparability between the models, the overall connection probability remains the same.

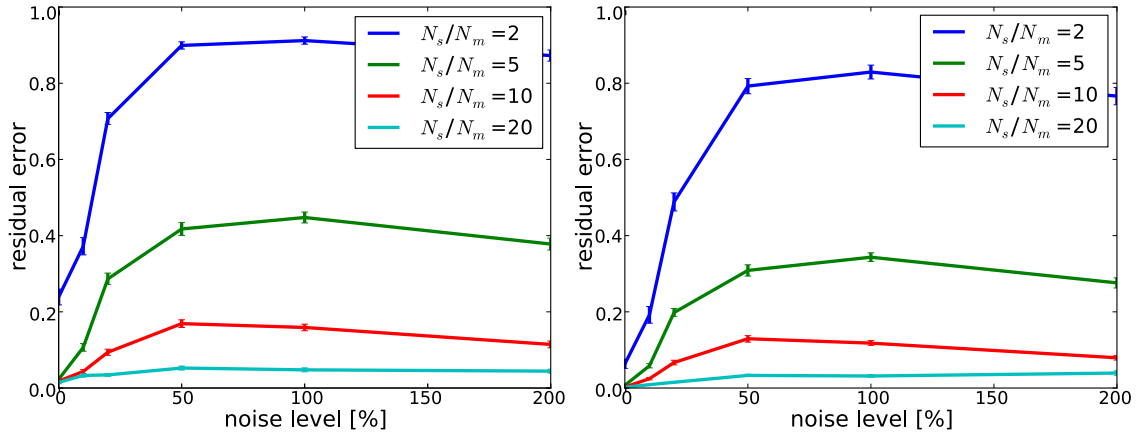


Figure 5.8: Residual error for different size ratios between sensory and motor populations in dependency of background noise level. Left: motor population, Right: sensory population. Noise has a higher impact on lower size ratios than on high size ratios. For high size ratios, the impact of noise is surprisingly small.

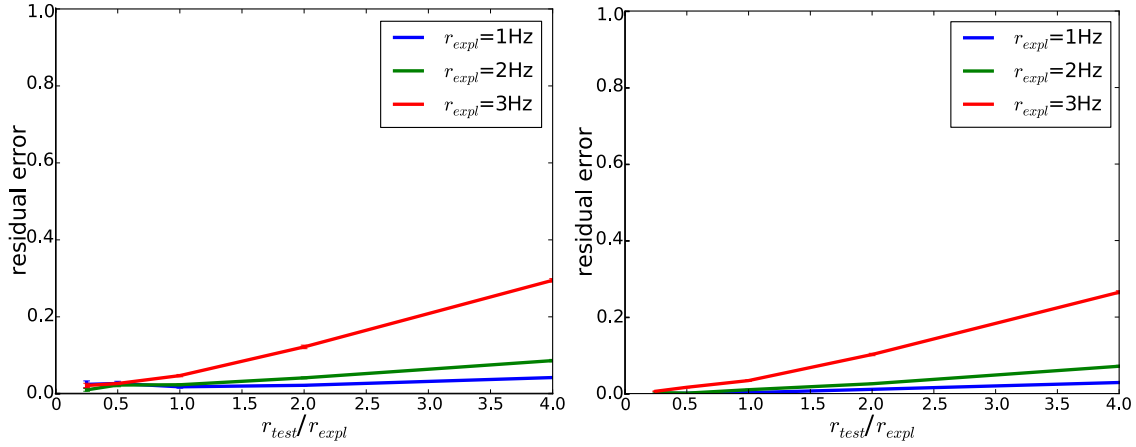
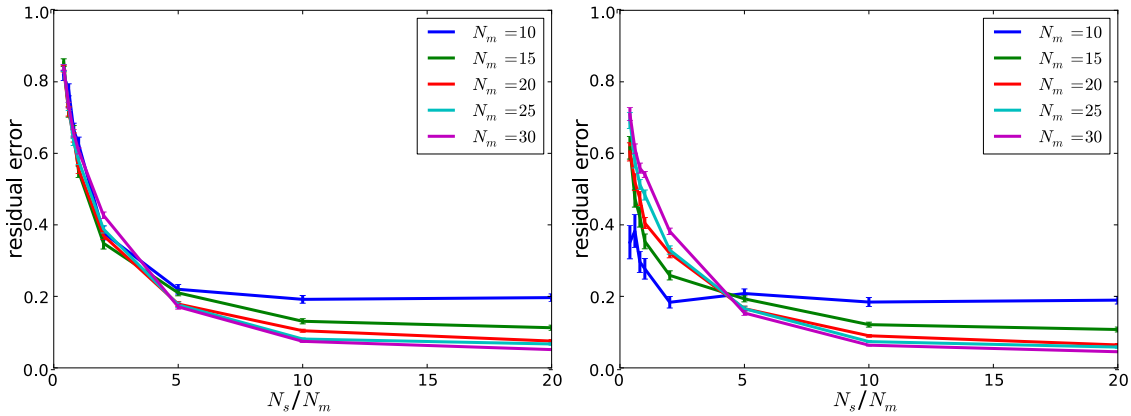
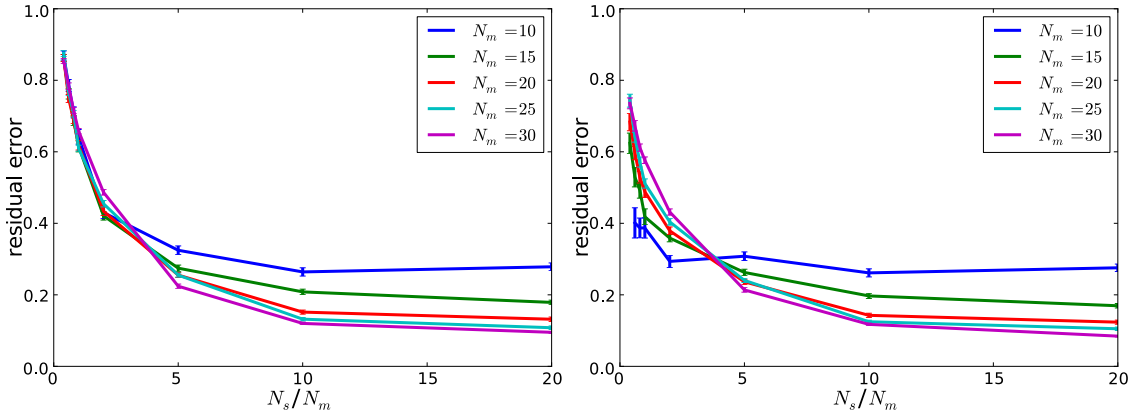


Figure 5.9: Residual error for different exploration firing rates in dependency on the firing rate ratios between song firing rate and exploration firing rate. Left: motor population, Right: sensory population. The residual error is low for low song firing rates for all exploration firing rate ratios and then rises to moderate level for higher ratios.



(a) Moderately complex world model with $N_w = 5$



(b) Moderately complex world model with $N_w = 10$

Figure 5.10: Residual error for different system sizes in dependency on size ratios between sensory and motor populations for a broad distribution of self-generated inputs with $N_w = 5$ resp. $N_w = 10$. Learning success is still relatively good with moderate residual errors. Left: motor population, Right: sensory population.

Experimentally, a high width of the self-generated input is equivalent to a broad distribution of loop delays, which is indeed found in experiments (e.g. [GKGH14], [SB06]).

To investigate how the complexity of the world model affects the learning success, the residual error for different system sizes and population ratios is computed for several different net widths of the self-generated input with $N_w = \{5, 10, 40\}$.

The residual errors for $N_w = 5$ resp. $N_w = 10$ are displayed in figure 5.10(a) resp. 5.10. Learning is no longer almost perfect, but still successful with moderate residual errors. While at low α the residual error is higher in larger system, it is smaller for larger systems for high α, where residual errors are lowest.

For the broadest distribution of self-generated input, $N_w = 40$, learning converges a lot more slowly, which is why $\eta_{norm} = 0.001$ was choosen. The residual error for different system sizes and population size ratios is displayed in figure 5.11. Here, learning breaks down completely with very high residual errors. Hence, learning of inverse models with this learning rule in spiking networks only works for relatively simple models of the world, but not for more complex pattern on pattern matching.

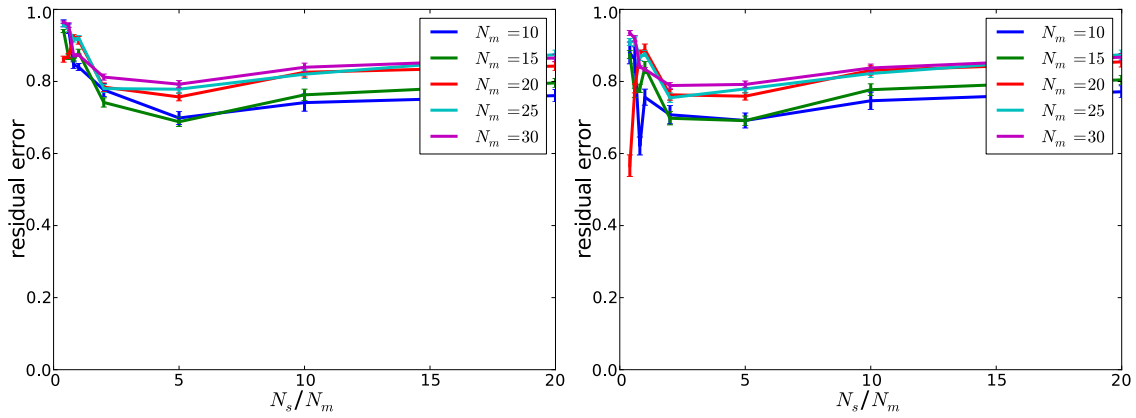


Figure 5.11: Residual error for different system sizes in dependency on size ratios between sensory and motor populations for a broad distribution of self-generated inputs $N_w = 40$. Learning success is very low with high residual errors. Left: motor population, Right: sensory population.

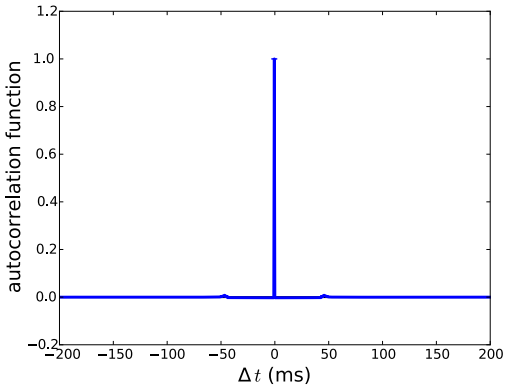
5.3.8 Experimentally testable Predictions: the Spike Autocorrelation

The long after-hyperpolarisation in the motor population in the model serves to avoid cyclic activity and to provide a testbed for self-generated input from the sensory population. A side effect of this hyperpolarisation is that the maximum firing rate in the motor population is limited, because immediately after a spike, each neuron is reset to a low membrane potential and thus has a very low spiking probability. This can be measured by measuring the autocorrelation of the spike trains. Figure 5.12 shows the autocorrelation of the spike trains in m for $N_m = 10$, $N_s = 100$, $r_{expl} = r_{song} = 7Hz$. There is a small, but noticeable dip in the autocorrelation of the length of the time of the loop delay. Furthermore, there is an upward bump in the autcorrelation function at the time of the self-generated input. This is a consequence of the fact that the membrane potential of the motor neuron rises to above resting potential due to that extra input. Note that this is a side-effect of the specific setup of the model, while the dip in the autocorrelation function is a generic property of the learning principle.
The autocorrelation could be accessible experimentally.

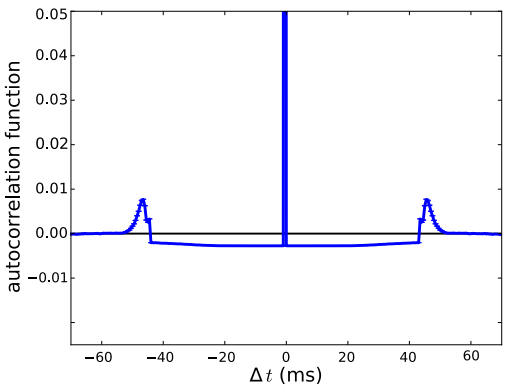
5.4 Discussion

In this chapter, I presented a simple and biologically plausible learning mechanism that can serve to learn inverse models in networks of spiking neurons.

The model predicts mirror neurons with a delay equivalent to the loop delay in the respective sensory and motor areas. This delay was linked to causal inverse models before ([HGH14]). In fact, experimental evidence for this type of delay was found in songbird brain area LMAN ([GKGH14]). In their study, [HGH14] suggest a simple hebbian learning rule, which relies on the comparison between self-generated and target input. However, there is no clear biological interpretation of how this comparison could be done. Additionally, in their study, to provide analytical tractability they only discuss a linear model



(a) Autocorrelation function of spiking activity in the motor population.



(b) Close-up of the autocorrelation function around $\Delta t = 0$ and $\rho(\Delta t) = 0$.

Figure 5.12: Autocorrelation function of the spiking activity in m. There is a dip of a width equivalent to the loop delay. The horizontal black line at autocorrelation 0 serves as a guide to the eye.

of sound generation and perception which is local in time. In the songbird, however, the process of sound generation and perception can be assumed to be both, non-linear and non-local in time. In this model, non-linear spiking neurons for sound perception and generation are included as well as a motor-sensory mapping that includes interactions not local in time.
The learning algorithm presented here is designed to operate on spiking neurons and is well able to invert non-linear models of the world. Furthermore, it can invert world models that spread the sensory input over some amount of time - up to about $N_w = 10$ learning works relatively well, which is equivalent to a temporal width of input into the sensory population of $10ms$. However, for broad distributions of time delays from motor to sensory population, learning breaks down.

In this chapter, simple integrate-and-fire neurons are used, because they capture the general properties of spiking neurons. However, to provide a large and long enough hyperpolarisation after each spike, a very low reset potential is chosen. While this is not biologically realistic, the learning process only relies on the fact that a substantial extra input around the time of the self-generated input from the sensory population back into the motor population is required to drive the membrane potential up to the plasticity threshold. In more realistic neuron models, such as the conductance-based integrate-and-fire-neuron model, this could be done with a more reasonable reset potential, in a way similar to what will be presented in chapter 7. Since these neuron models are much closer to real neurons than the simple integrate-and-fire neuron used here, it can be assumed that in the songbird no large visible hyperpolarisation is required.

In fact, the learning only relies on the fact that all synapses from sensory to motor population are strengthened, when coactive, until the sensory input is large enough to reach a plasticity threshold below the spiking threshold. For learning to be successful, the membrane potential needs to be at a sufficient distance from the plasticity threshold at the

time of the self-generated input (while the amplitude of the self-generated input input is still small - after learning, the self-generated input pushes the membrane potential towards the plasticity threshold). The mechanism suggested here is not the only mechanism that can accomplish that: An adaptive spiking threshold, that is raised with each motor spike and decays exponentially with the plasticity threshold at a fixed distance to the spiking threshold would have the same learning effect. The interplay between this type of adaptive threshold and STDP was shown to be beneficial for associative learning before [DLH10]. Hence, the hyperpolarisation does not need to be as pronounced as it is here, or, in fact, present at all, but any such mechanism will leave a trace in the spiking statistics of the motor neuron: There will be a dip in the autocorrelation function of the duration of the loop delay (compare figure 5.12). Hence, this dip in the autocorrelation function is a trademark of this learning principle, which might be experimentally accessible.

The other part of the learning principle, RSTDP, has been observed experimentally in several systems, either in the direct form of RSTDP on excitatory synapses [FPD05, SH06] or in the indirect form of regular STDP on inhibitory synapses [HNA06], which has the same net effect on the membrane potential. Several studies investigate the effect of STDP on network function, however mostly with a focus on stability issues (e.g. [SMA00, ID03, VSZ+11]). In this model, however, it is shown that it can serve a constructive role in learning inverse models.

There are several areas in the avian song system that are candidate areas for both, motor and sensory population. Mirror neurons of the time delay in the order of magnitude of the loop delay have only been found in LMAN [GKGH14], which makes LMAN the most likely candidate for the motor area. However, firing rates in LMAN are generally higher than the very low firing rates required in this very simple model with typical firing rates at between $12Hz$ and $40Hz$ [ÖAF05, Leo04]. In the model, not only does the learning success decrease for higher firing rates, firing rates are limited by the long hyperpolarisation in the motor neurons. This effect is generic to the learning principle and would occur as well, if it was implemented by a spike frequence adaptation as in [DLH10].
One candidate area for a sensory area involved in song production is HVC. However, there are no direct connections from HVC to LMAN (or from any other primarily sensory area), which shows that the very simple setup presented here is a strong simplification of the much more complex real system.
However, HVC is not only a sensory area, but also involved as a motor area in song production. Hence, it is also a candidate area for the motor area of this model. Firing rates in HVC are very low [HKF02], which would match the low firing rates required for optimal learning in the model. However, the only mirror neurons found in HVC have zero time delay between song and playback [PPNM08].

To allow for the inversion of any type of forward mapping, a size difference between the involved neural populations is advantageous. In the model, learning works particularly well, if the sensory area is at least five times bigger than the motor area. However, it is unlikely that this size difference would be reflected in the size of the brain areas in the songbird.
Learning success depends very little on system size, if learning is successful. When learning is only moderately successful - in the case of more complex world models or high firing

rates - larger systems are advantageous. However, when learning breaks down for very broad distributions of self-generated input, larger system sizes don't help to alleviate this effect. This is particularly important, because songbird brain areas are much larger than the up to $N_m = 30$ motor neurons considered here.

Songbirds learn to sing not in isolation, but rather in large bird colonies. This raises the question, how well the model bird can learn in the presence of background noise. To investigate this, the residual error was measured in dependency of the noise level in percent of the average activity of the tutor song. While there are substantial difficulties for low size ratios between the motor and sensory population α, learning is surprisingly robust under the influence of noise, with little influence for large size ratios between motor and sensory area ranging up to 200%.

In this chapter, a new mechanism for the learning of inverse models in a simple setup was suggested, that consists of biologically plausible parts and works on spiking neurons. While it is unclear, if this exact setup can be found in the songbird, it is a conceptually interesting approach to inverse model learning. However, the role of zero delay mirror neurons and possible mechanisms for the inversion of more realistic world models remain unclear - they will be discussed again in chapter 8.

Chapter 6

The Perceptron Learning Rule

While in the last chapter the application of Anti-Hebbian spike-timing-dependent plasticity (RSTDP) to unsupervised learning of inverse models was discussed, here the focus is set on the general properties of RSTDP in a much simpler network architecture: the single layer feed-forward network. RSTDP is able to perform Perceptron learning very similarly to the original Perceptron Learning Rule as introduced by [Ros58] and below. In this chapter, it is proven that RSTDP of excitatory synapses (or CSTDP on inhibitory synapses) when acting in concert with neuronal after-hyperpolarisation and depolarization dependent LTD is sufficient for realizing the classical Perceptron learning rule. The neuron model and the plasticity rule in this chapter are the same as in chapter 5, but will be repeated for completeness.

The results from this chapter were obtained in collaboration with Christian Albers and published in [AWP13] together with a quantitative assessment of the learning capabilities of the learning rule for the associative learning of output spikes in response to input patterns elongated in time, both for precisely timed output spike (Chronotron problem) and for output spikes without timing restrictions (Tempotron problem).

6.1 Introduction

Perceptrons are paradigmatic building blocks of neural networks [HKP91] (see section 3.4.2.1 for details). The original Perceptron Learning Rule (PLR) is a supervised learning rule that employs a threshold to control weight changes, which also serves as a margin to enhance robustness [Ros58, MP69]. If the learning set is linearly separable, the PLR algorithm is guaranteed to converge in a finite number of steps [HKP91], which justifies the term 'perfect learning'.

Associative learning can be considered a special case of supervised learning, where the activity of the output neuron is used as a teacher signal such that after learning missing activities are filled in. For this reason the PLR is very useful for building associative memories in recurrent networks, where it can serve to learn arbitrary patterns in a 'quasi-unsupervised' way. Here it turned out to be far more efficient than the simple Hebb rule, leading to a superior memory capacity and non-symmetric weights [DO87]. Note also that over-learning from repetitions of training examples is not possible with the PLR because weight changes vanish as soon as the accumulated inputs are sufficient, a property

which in contrast to the naïve Hebb rule makes it suitable also for incremental learning of associative memories from sequential presentation of patterns.

On the other hand, it is not known if and how biological synaptic mechanisms might realize the success-dependent self-regulation of the PLR in networks of spiking neurons in the brain. For example in the Tempotron [GS06], a generalization of the Perceptron to spatio-temporal patterns, learning was conceived even somewhat less biological than the PLR, since here it not only depends on the potential classification success, but also on a process that is not local in time, namely the localization of the absolute maximum of the (virtual) postsynaptic membrane potential of the postsynaptic neuron. The simplified Tempotron learning rule, while biologically more plausible, still relies on a reward signal which tells each neuron specifically whether it should have spiked or not. Taken together, while highly desirable, the feature of self regulation in the PLR still poses a challenge for biologically realistic synaptic mechanisms.

The classical form of spike-timing-dependent plasticity (STDP) for excitatory synapses (here denoted CSTDP) states that the causal temporal order of first presynaptic activity and then postsynaptic activity leads to long-term potentiation of the synapse (LTP) while the reverse order leads to long-term depression (LTD)[DP04, DP06, CD08]. More recently, however, it became clear that STDP can exhibit different dependencies on the temporal order of spikes. In particular, it was found that the reversed temporal order (first post- then presynaptic spiking) could lead to LTP (and vice versa; RSTDP), depending on the location of the synapse on the dendrite [FPD05, SH06]. For inhibitory synapses some experiments were performed which indicate that here STDP exists as well and has the form of CSTDP [HNA06]. Note that CSTDP of inhibitory synapses is equivalent to RSTDP of excitatory synapses in its effect on the postsynaptic neuron. Additionally it has been shown that CSTDP does not always rely on spikes, but that strong subthreshold depolarization can replace the postsynaptic spike for LTD while keeping the usual timing dependence [STN04]. Therefore, here a second threshold for the induction of timing dependent LTD is assumed. For simplicity and without loss of generality, investigations can be restricted to RSTDP for synapses that in contradiction to Dale's law can change their sign.

It is very likely that plasticity rules and dynamical properties of neurons co-evolved to yield maximally beneficial interplay. Combining them could reveal new and desirable effects. A modelling example for a beneficial effect of such an interplay was investigated in [DLH10], where CSTDP interacted with spike-frequency adaptation of the postsynaptic neuron to perform a gradient descent on a square error. Several other studies investigate the effect of STDP on network function, however mostly with a focus on stability issues (e.g. [SMA00, ID03, VSZ$^+$11]). In contrast, here focus is put on the constructive role of STDP for associative learning. It is proven that RSTDP of excitatory synapses (or CSTDP of inhibitory synapses) when acting in concert with neuronal after-hyperpolarisation and depolarization-dependent LTD is sufficient for realizing the classical Perceptron learning rule.

6.2 The Model

6.2.1 Neuron Model and Network Structure

Assume a feed-forward network of N presynaptic neurons and one postsynaptic integrate-and-fire neuron with a membrane potential U governed by

$$\tau_U \dot{U} = -U + I_{syn} + I_{ext}, \tag{6.1}$$

where I_{syn} denotes the input from the presynaptic neurons, and I_{ext} is an input which can be used to drive the postsynaptic neuron to spike at certain times. When the neuron reaches a threshold potential U_{thr}, it is reset to a reset potential $U_{reset} < 0$, from where it decays back to the resting potential $U_\infty = 0$ with time constant τ_U. Spikes and other signals (depolarization) take finite times to travel down the axon (τ_a) and the dendrite (τ_d). Synaptic transmission takes the form of delta pulses, which reach the soma of the postsynaptic neuron after time $\tau_a + \tau_d$, and are modulated by the synaptic weight w. The presynaptic spike train of neuron i is denoted as x_i with spike times t^i_{pre}:

$$x_i(t) = \sum_{t^i_{pre}} \delta(t - t^i_{pre}). \tag{6.2}$$

A postsynaptic neuron receives the input $I_{syn}(t) = \sum_i w_i x_i(t - \tau_a - \tau_d)$. The postsynaptic spike train is similarly denoted by $y(t) = \sum_{t_{post}} \delta(t - t_{post})$.

6.2.2 The Plasticity Rule

The plasticity rule employed here mimics reverse STDP: A postsynaptic spike which arrives at the synapse shortly before a presynaptic spike leads to synaptic potentiation. For synaptic depression the relevant signal is not the spike, but the point in time t_k where $U(t)$ crosses an additional threshold U_{st} from below, with $U_\infty < U_{st} < U_{thr}$ ("subthreshold threshold"). These events are modelled as δ-pulses in the function $z(t) = \sum_k \delta(t - t_k)$ (see figure 6.1 (a) for an illustration of the principle). The temporal characteristic of (reverse) STDP is preserved: If a presynaptic spike occurs shortly before the membrane potential crosses this threshold, the synapse depresses. Such timing-dependent LTD without postsynaptic spiking has been observed, although with classical timing requirements [STN04].

This is formalized by letting pre- and postsynaptic spikes each drive a synaptic trace $\bar{x}(t)$ resp. $\bar{y}(t)$:

$$\begin{aligned} \tau_{pre}\dot{\bar{x}} &= -\bar{x} + x(t - \tau_a) \\ \tau_{post}\dot{\bar{y}} &= -\bar{y} + y(t - \tau_d). \end{aligned} \tag{6.3}$$

The learning rule is a read–out of the traces by spikes and threshold crossing events, respectively:

$$\dot{w} \propto \bar{y}x(t - \tau_a) - \gamma\bar{x}z(t - \tau_d), \tag{6.4}$$

where γ is a factor which scales depression and potentiation relative to each other. Figure 6.1 (b) shows how this plasticity rule creates RSTDP.

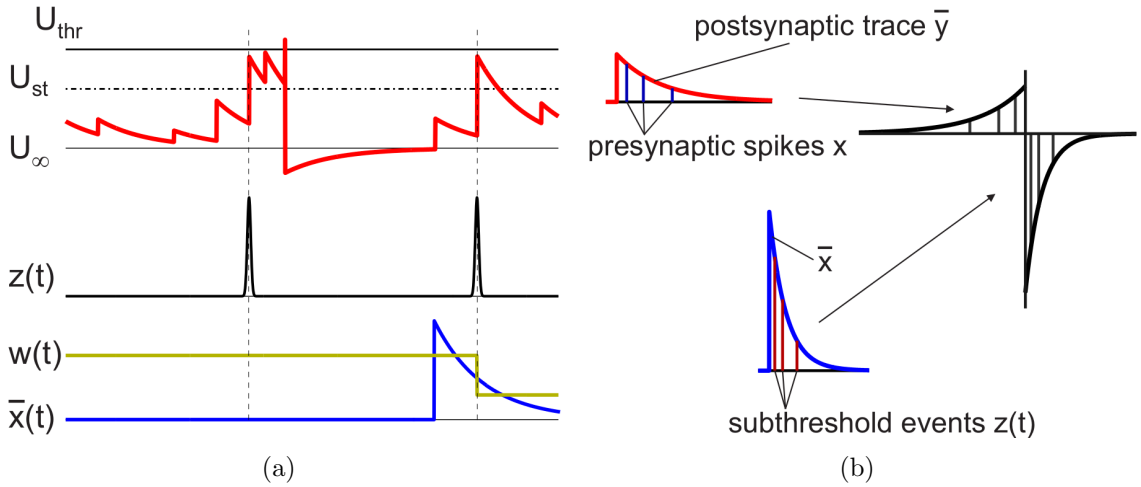


Figure 6.1: Illustration of STDP mechanism. (a) Upper trace (red) is the membrane potential of the postsynaptic neuron. Shown are the firing threshold U_{thr} and the threshold for LTD U_{st}. Middle trace (black) is the variable $z(t)$, the train of LTD threshold crossing events. Please note that the first spike in $z(t)$ occurs at a different time than the neuronal spike. Bottom traces show $w(t)$ (yellow) and $\bar{x}$ (blue) of a selected synapse. The second event in z reads out the trace of the presynaptic spike $\bar{x}$, leading to LTD. (b) Learning rule (6.4) is equivalent to RSTDP. A postsynaptic spike leads to an instantaneous jump in the trace $\bar{y}$ (top left, red line), which decays exponentially. Subsequent presynaptic spikes (dark blue bars and corresponding thin gray bars in the STDP window) "read" out the state of the trace for the respective $\Delta t = t_{pre} - t_{post}$. Similarly, $z(t)$ reads out the presynaptic trace $\bar{x}$ (lower left, blue line). Sampling for all possible times results in the STDP window (right).

6.3 Equivalence to Perceptron Learning Rule

The Perceptron Learning Rule (PLR) for positive binary inputs and outputs is given by

$$\Delta w_i^\mu \propto x_0^{i,\mu}(2y_0^\mu - 1)\Theta\left[\kappa - (2y_0^\mu - 1)(h^\mu - \vartheta)\right], \tag{6.5}$$

where $x_0^{i,\mu} \in \{0,1\}$ denotes the activity of presynaptic neuron i in pattern $\mu \in \{1,\dots,P\}$, $y_0^\mu \in \{0,1\}$ signals the desired response to pattern μ, $\kappa > 0$ is a margin which ensures a certain robustness against noise after convergence, $h^\mu = \sum_i w_i x_0^{i,\mu}$ is the input to a postsynaptic neuron, ϑ denotes the firing threshold, and $\Theta(x)$ denotes the Heaviside step function. If the P patterns are linearly separable, the Perceptron will converge to a correct solution of the weights in a finite number of steps. For random patterns this is generally the case for $P < 2N$. A finite margin κ reduces the capacity.

Interestingly, for the case of temporally well separated synchronous spike patterns the combination of RSTDP-like synaptic plasticity dynamics with depolarization-dependent LTD and neuronal hyperpolarization leads to a plasticity rule which can be mapped to the Perceptron Learning Rule. To cut down unnecessary notation in the derivation, the indices i and μ are dropped except where necessary and the range of considered times is limited to $0 \le t \le \tau_a + 2\tau_d$.

Consider a single postsynaptic neuron with N presynaptic neurons, with the condition $\tau_d < \tau_a$. During learning, presynaptic spike patterns consisting of synchronous spikes at time $t = 0$ are induced, concurrent with a possibly occuring postsynaptic spike which signals the class the presynaptic pattern belongs to. This is equivalent to the setting of a single layered Perceptron with binary neurons. With x_0 and y_0 used as above, the pre- and postsynaptic activity can be written as $x(t) = x_0\delta(t)$ and $y(t) = y_0\delta(t)$. The membrane potential of the postsynaptic neuron depends on y_0:

$$\begin{aligned} U(t) &= y_0 U_{reset}\exp(-t/\tau_U) \\ U(\tau_a+\tau_d) &= y_0 U_{reset}\exp(-(\tau_a+\tau_d)/\tau_U) = y_0 U_{ad}. \end{aligned} \tag{6.6}$$

Similarly, the synaptic current is

$$\begin{aligned} I_{syn}(t) &= \sum_i w_i x_0^i \delta(t-\tau_a-\tau_d) \\ I_{syn}(\tau_a+\tau_d) &= \sum_i w_i x_0^i = I_{ad}. \end{aligned} \tag{6.7}$$

The activity traces at the synapses are

$$\begin{aligned} \bar{x}(t) &= x_0\Theta(t-\tau_a)\frac{\exp(-(t-\tau_a)/\tau_{pre})}{\tau_{pre}} \\ \bar{y}(t) &= y_0\Theta(t-\tau_d)\frac{\exp(-(t-\tau_d)/\tau_{post})}{\tau_{post}}. \end{aligned} \tag{6.8}$$

The variable of threshold crossing $z(t)$ depends on the history of the postsynaptic neurons, which again can be written with the aid of y_0 as

$$z(t) = \Theta(I_{ad} + y_0 U_{ad} - U_{st})\delta(t-\tau_a-\tau_d). \tag{6.9}$$

Here, Θ reflects the condition for induction of LTD. Only when the postsynaptic input at time $t = \tau_a + \tau_d$ is greater than the residual hyperpolarization ($U_{ad} < 0$!) plus the

threshold U_{st}, a potential LTD event is registered. These are the ingredients for the plasticity rule (6.4):

$$\begin{aligned}\Delta w \propto & \int \left[\bar{y}x(t-\tau_a) - \gamma\bar{x}z(t-\tau_d)\right] dt \\ =& x_0 y_0 \frac{\exp(-(\tau_a+\tau_d)/\tau_{post})}{\tau_{post}} - \gamma x_0 \frac{\exp(-2\tau_d/\tau_{pre})}{\tau_{pre}} \Theta(I_{ad} + y_0 U_{ad} - U_{st}).\end{aligned} \tag{6.10}$$

This expression can be shortened by choosing γ such that the factors of both terms are equal, which can be dropped subsequently:

$$\Delta w \propto x_0 \left(y_0 - \Theta(I_{ad} + y_0 U_{ad} - U_{st})\right). \tag{6.11}$$

Expanding equation 6.11 by adding and substracting $y_0\Theta(I_{ad} + y_0 U_{ad} - U_{st})$ gives

$$\begin{aligned}\Delta w \propto & x_0 \left[y_0(1 - \Theta(I_{ad} + y_0 U_{ad} - U_{st})) - (1-y_0)\Theta(I_{ad} + y_0 U_{ad} - U_{st})\right] \\ =& x_0 \left[y_0 \Theta(-I_{ad} - U_{ad} + U_{st}) - (1-y_0)\Theta(I_{ad} - U_{st})\right].\end{aligned} \tag{6.12}$$

In the last transformation, $1 - \Theta(x) = \Theta(-x)$ was used and y_0 was dropped from the argument of the Heaviside functions, as the two terms are separated into the two cases $y_0 = 0$ and $y_0 = 1$. A similar transformation can be done to construct an expression G which is used to substitute the arguments of the Heaviside functions. That expression is

$$G = I_{ad} - U_{st} + y_0(-2I_{ad} - U_{ad} + 2U_{st}), \tag{6.13}$$

with which the arguments are replaced

$$\Delta w \propto x_0 y_0 \Theta(G) - x_0(1-y_0)\Theta(G) = x_0(2y_0 - 1)\Theta(G). \tag{6.14}$$

The last task is to show that G and the argument of the Heaviside function in equation (6.5) are equivalent. For this, $I_{ad} = h$, $U_{ad} = -2\kappa$ and $U_{st} = \vartheta - \kappa$ is chosen. Note that $\vartheta = U_{thr}$ is the firing threshold. Introducing this into G yields

$$\begin{aligned}G =& I_{ad} - U_{st} + y_0(-2I_{ad} - U_{ad} + 2U_{st}) \\ =& h - \vartheta + \kappa + 2y_0 h + 2y_0\kappa + 2y_0\vartheta - 2y_0\kappa \\ =& \kappa - (2y_0 - 1)(h - \vartheta),\end{aligned} \tag{6.15}$$

which is the same as the argument of the Heaviside function in equation (6.5). Therefore, equivalence of the two learning rules was shown.

6.4 Discussion

In this chapter, the biologically highly plausible approach to learning in neuronal networks introduced in chapter 5 is applied to learning in feed-forward networks. RSTDP with subthreshold LTD in concert with hyperpolarisation is shown to be mathematically equivalent to the Perceptron learning rule for activity patterns consisting of synchronous spikes, thereby inheriting the highly desirable properties of the PLR (convergence in finite time, stop condition if performance is sufficient and robustness against noise). This provides a biologically plausible mechanism to build associative memories with a capacity

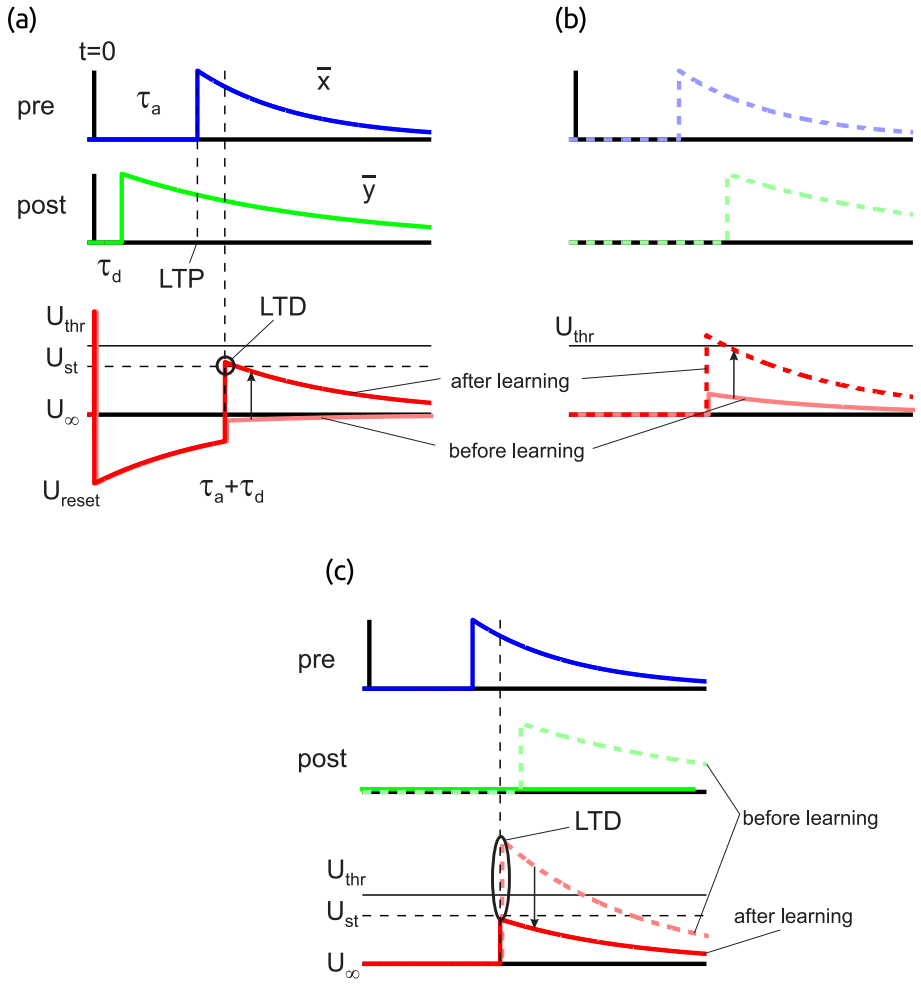


Figure 6.2: Illustration of Perceptron learning with RSTDP with subthreshold LTD and postsynaptic hyperpolarization. Shown are the traces $\bar{x}$, $\bar{y}$ and U. Pre- and postsynaptic spikes are displayed as black bars at $t = 0$. (a) Learning in the case of $y_0 = 1$, i.e. a postsynaptic spike as the desired output. Initially the weights are too low and the synaptic current (summed PSPs) is smaller than U_{st}. Weight change is LTP only until the membrane potential hits U_{st} during pattern presentation. At this point LTP and LTD exactly cancel each other out, and learning stops. (b) Pattern completion for $y_0 = 1$. Shown are the same traces as in A at the absence of an inital postsynaptic spike. The membrane potential after learning is drawn as a dashed line to highlight the amplitude. Without the initial hyperpolarization, the synaptic current after learning is large enough to cross the spiking threshold and the postsynaptic neuron fires the desired spike. Learning until U_{st} is reached ensures a minimum height of synaptic currents and therefore robustness against noise. (c) Pattern presentation and completion for $y_0 = 0$. Initially, the synaptic current during pattern presentation causes a spike and consequently LTD. Learning stops when the membrane potential stays below U_{st}. Again, this ensures a certain robustness against noise, analogous to the margin in the PLR.

close to the theoretical maximum. Equivalence of STDP with the PRL was shown before in [LNM05], but this equivalence only holds on average. Here, in contrast, a novel approach is presented that ensures exact mathematical equivalence to the PRL.

The mechanism proposed here is complementary to a previous approach [DLH10] which uses CSTDP in combination with spike frequency adaptation to perform gradient descent learning on a squared error. However, that approach relies on an explicit teacher signal, and is not applicable to auto-associative memories in recurrent networks. Most importantly, the approach presented here inherits the important feature of self-regulation and fast convergence from the original Perceptron which is absent in [DLH10].

Spike after-hyperpolarization is often neglected in theoretical studies or assumed to only play a role in network stabilization by providing refractoriness. Depolarization dependent STDP receives little attention in modelling studies (but see [CBVG10]), possibly because there are only few studies which show that such a mechanism exists [STN04, FDV09]. The novelty of the learning mechanism presented here lies in the constructive roles both play in concert. After-hyperpolarization allows synaptic potentiation for presynaptic inputs immediately after the teacher spike without causing additional non-teacher spikes, which would be detrimental for learning. During recall, the absence of the hyperpolarization ensures the then desired threshold crossing of the membrane potential (see figure 2 (b)). Subthreshold LTD guarantees convergence of learning. It counteracts synaptic potentiation when the membrane potential becomes sufficiently high after the teacher spike. The combination of both provides the learning margin, which makes the resulting network robust against noise in the input. Taken together, these results show that the interplay of neuronal dynamics and synaptic plasticity rules can give rise to powerful learning dynamics.

Chapter 7

Chronotron Learning with Membrane Potential Dependent Plasticity

In chapters 5 and 6, the learning of classification of input spike patterns that are purely spatial was investigated in the context of learning of simple inverse models and on the example of Perceptron learning. This learning principle consists of combining a strong hyperpolarisation with a synaptic learning rule that changes synapses to raise the membrane potential until a plasticity threshold below the firing threshold is reached. So far, this synaptic plasticity rule was chosen to be Anti-Hebbian Spike-Timing-Dependent Plasticity. Here, the learning capabilities of a different synaptic learning rule will be investigated: Membrane Potential Dependent Plasticity (MPDP), which strives to balance excitatory and inhibitory inputs to keep the membrane potential bounded. The combination of this learning rule with hyperpolarisation provides a learning principle that is not only biologically plausible, but also enables a neuron to learn to spike at precisely defined times in response to spatio-temporal input patterns (Chronotron problem, see 3.4.2.3).

7.1 Introduction

Precise and recurring spatio-temporal patterns of action potentials are observed in various biological neuronal networks. In zebra finches, precise sequences of activations in region HVC are found during singing and listening to the own song [HKF02]. Also, when spike times of sensory neurons are measured, the variability of latencies relative to the onset of an externally induced stimulus is often higher than if the latencies are measured relative to other sensory neurons [GM08, Mas13]; spike times covary. This allows to conclude that information about the stimulus is coded in spatio-temporal spike patterns. Theoretical considerations show that in some situations spike-time coding is superior to rate coding [VT01]. Xu and colleagues demonstrated that through associative training it is possible to imprint new sequences of activations in visual cortex [XJPD12], which shows that there are plasticity mechanisms which are used to learn precise sequences.
These observations suggest that spatio-temporal patterns of spike activities underlie coding and processing of information in many networks of the brain. However, it is not known which synaptic plasticity mechanisms enable neuronal networks to learn, generate,

and read out precise action potential patterns. A theoretical framework to investigate this question is the Chronotron, where the postsynaptic neuron is trained to fire a spike at predefined times relative to the onset of a fixed input pattern [Flo12a]. A natural candidate plasticity rule for Chronotron training is Spike-Timing Dependent Plasticity (STDP) [CD08] in combination with a supervisor who enforces spikes at the desired times. Legenstein and colleagues [LNM05] investigated the capabilities of supervised STDP in the Chronotron task and identified a key problem: STDP has no means to distinguish between desired spikes caused by the supervisor and spurious spikes resulting from the neuronal dynamics. As a result, every spike gets reinforced, and plasticity does not terminate when the correct output is achieved, which eventually leads to unlearning of the desired synaptic state. The failings of STDP hint at the requirements of a working learning algorithm. Information about the type of a spike (desired or spurious) has to be available to each synapse, where it modulates spike time based synaptic plasticity. Synapses evoking undesired spikes should be weakened, synapses that contribute to desired spikes should be strengthened, but only until the self-generated output activity matches the desired one. Plasticity should cease if the output neurons generate the desired spikes without supervisor intervention. In other words, at the core of a learning algorithm needs to be a comparison of actual and target activity, and synaptic changes have to be computed based on the difference between the two.

In recent years, a number of supervised learning rules have been proposed to train to fire temporally precise output spikes in response to recurring spatio-temporal input patterns [BKLP02, Flo12a, MRÖS14]. They compare the target spike train to the self-generated (actual) output and devise synaptic changes to transform the latter into the former. However, because spikes are discrete events in time that influence the future dynamics of the neuron, the comparison is necessarily non-local in time, which might be difficult to implement for a biological neuron and synapse. Another group of algorithms performs a comparison of actual and target firing rate instead of spike times [PK10, XS04, BSP13, US14]. Because they work with the instantaneous firing rate, they do not rely on sampling of discrete spikes and therefore the comparison is local in time. It is interesting to note that these learning algorithms are implicitely sensitive to the current membrane potential, of which the firing rate is a monotonous function. However, two important questions remain unanswered: How is the desired activity communicated to a biological neuron and how does the synapse compute the difference?

In this chapter, the learning capabilities of a plasticity rule, which relies only on postsynaptic membrane potential and presynaptic spikes as signals, are investigated. To distinguish it from spike times based rules, it is called Membrane Potential Dependent Plasticity (MPDP). MPDP is derived from a homeostatic requirement on the voltage and it is shown that in combination with spike after-hyperpolarisation (SAHP) it is compatible with experimentally observed STDP of inhibitory synapses [HNA06]. Despite its Anti-Hebbian nature, MPDP combined with SAHP can be used to train a neuron to generate desired temporally structured spiking output in an associative manner. During learning, the supervisor or teacher induces spikes at the desired times by a strong input. Because of the differences in the time course of the voltage, a synapse can sense the difference between spurious spikes caused by weak inputs and teacher spikes caused by strong inputs. As a consequence, weight changes are matched to the respective spike type. Therefore, the learning algorithm provides a biologically plausible answer for the open question presented above.

This learning algorithm was quantitatively assessed in [AWP16], where the memory capacity and the noise tolerance were tested in the most simple neuron model presented here, the integrate-and-fire neuron.

7.2 The Model

In this section, the models used are presented. First, the simpler leaky integrate-and-fire neuron model (LIF neuron) is introduced and used to derive the MPDP rule. Second, MPDP is applied to a more realistic conductance based integrate-and-fire neuron and a Hodgkin-Huxley type neuron.

7.2.1 Neuron Models

Three different neuron models are introduced to show that the learning algorithm suggested here is capable of learning precise spike times in response to spatio-temporal input patterns irrespective of the specifics of the neuron model.

7.2.1.1 The simple Integrate-and-Fire Neuron

The model for the simple integrate-and-fire neuron is the same as in chapter 5 and 6, however, the description will be repeated here for completeness.
Plasticity processes are investigated in several different network setups. In each of these setups, a neuron j receives input from other neurons i via plastic synapses. t_i^k is the time of the k-th spike of presynaptic neuron with index i. The neuron is modelled as a simple leaky integrate-and-fire neuron. The formulation of the SRM_0 facilitates the derivation of the plasticity rule ([GK02]). The neuronal voltage $V(t)$ is given by the sum of weighted synaptic input kernels $\varepsilon(s)$ (postsynaptic potentials, PSPs) and reset kernels $R(s)$, which model the neuronal reset after a spike. External input currents $I_{ext}(t)$ are low-pass filtered with a response kernel $\kappa(s)$. The full equation reads

$$V(t) = \sum_i w_i \sum_k \varepsilon(t - t_k^i - t_{delay}) + \sum_{t_j} R(t - t_j) + \int_0^\infty \kappa(t-s) I_{ext}(s) ds. \tag{7.1}$$

Here, w_i is the weight from presynaptic neuron j to the postsynaptic neuron. $\kappa = \exp\left(-(t-s)/\tau_m\right)$ is the passive response kernel by which external currents are filtered. A delay of synaptic transmission t_{delay} is included. The other kernels are

$$\begin{aligned} \varepsilon(s) &= \Theta(s) \frac{1}{\tau_m - \tau_s} \left(\exp(-s/\tau_m) - \exp(-s/\tau_s)\right) \\ R(s) &= \Theta(s)(V_{reset} - V_{thr}) \exp(-s/\tau_m). \end{aligned} \tag{7.2}$$

$\tau_m = 8ms$ is the membrane time constant of a LIF neuron determining the decay of voltage perturbations, and $\tau_s = 2ms$ is the decay time constant of synaptic currents, which defines the rise time of the PSP kernel. $\Theta(s)$ is the Heaviside step function. If there is no input, the voltage relaxes back to $V_{eq} = 0$. Spiking in this model is deterministic: If $V(t') = V_{thr} = 20mV$, the neuron spikes and a reset kernel is added at time $t' = t_{post}$. The formulation of the kernel makes sure that the voltage is always reset to $V_{reset} = -60mV < V_{eq}$.

7.2.1.2 The conductance-based Integrate-and-Fire Neuron

The simple model above suffers from the fact that MPDP is agnostic to the type of synapse. In principle, MPDP can turn excitatory synapses into inhibitory ones by changing the sign of any weight w_i. Since this is a violation of Dale's law, a more biologically realistic scenario involving MPDP is presented.
The presynaptic population is split into N_{ex} excitatory and N_{in} inhibitory neurons. The postsynaptic neuron is modelled as a conductance based LIF neuron governed by

$$C_m \frac{\mathrm{d}V}{\mathrm{d}t} = -g_L(V - V_L) - (g_{sl} + g_f)(V - V_h) - g_{ex}(V - V_{ex}) - g_{in}(V - Vin) \ , \qquad (7.3)$$

where V denotes the membrane potential, $C_m = 0.16\mu F$ the membrane capacitance, $V_L = -70mV$ the resting potential, $g_L = 20$ the leak conductance, $V_i = -75mV$ and $V_{ex} = 0$ the reversal potential of inhibition and excitation, respectively and g_{in} and g_{ex} their respective conductances. The spike after-hyperpolarisation is modelled to be biphasic consisting of a fast and a slow part, described by conductances g_f and g_{sl} that keep the membrane potential close to the hyperpolarisation potential $V_h = V_i$. When the membrane potential surpasses the spiking threshold $V_{thr} = -50$ at time t_{post}, a spike is registered and the membrane potential is reset to $V_{reset} = V_h$. All conductances are modelled as step and decay functions. The reset conductances are given by

$$\tau_{f,sl}\dot{g}_{f,sl} = -g_{f,sl} + \Delta g_{f,sl} \sum_{t_{post}} \delta\left(t - t_{post}\right) \ , \qquad (7.4)$$

where $\Delta g_{sl} = 5$ resp. $\Delta g_{sl} = 1000$ is the increase of the fast and slow conductance at the time of each postsynaptic spike. They decay back with time constants $\tau_f = \tau_s < \tau_{sl} = C_m/g_L$. The input conductances g_{ex} and g_{in} are step and decay functions as well, which are increased by w_i when presynaptic neuron i spikes and decay with time constant $\tau_s = 2ms$. w_i denotes the strength of synapse i.

7.2.1.3 The Hodgkin-Huxley-type Neuron

To test whether the learning mechanism suggested below is also capable of learning spike associations for neuron models that generate spikes, it is tested on a Hodgkin-Huxley type neuron (see section 3.1.2.3).
Inhibitory and excitatory inputs remain separated into two input populations. They provide input into the output neuron with membrane potential $V(t)$ given by

$$C_m\dot{V} = -g_L(V - V_L) - g_K n^4(V - V_K) - g_{Na}m^3h(V - V_{Na}) - g_{ex}(V - V_{ex}) - g_{in}(V - Vin) \qquad (7.5)$$

where $V_L = -65mV$ is the leak potential, $g_L = 0.1mS/cm$ is the leak conductance, g_{ex} resp. g_{in} are the conductance governing excitatory resp. inhibitory input from the input populations and $V_{ex} = 0mV$ resp. $V_{in} = -75mV$ are their reversal potentials. $V_{Na} = 55mV$ is the reversal potential of sodium, $V_K = -90mV$ is the reversal potential of potassium, $C_m = 1\mu F/cm^2$ is the membrane capacitance, $g_K = 9mS/cm^2$ is the maximum potassium conductance and $g_{Na} = 35mS/cm^2$ is the maximum sodium conductance.
The conductance variables $n(t, V)$, $m(t, V)$ and $h(t, V)$ are time and voltage dependent

and take values between 0 and 1. They are given by

$$\dot{n} = \alpha_n(V)(1-n) - \beta_n(V)n \tag{7.6}$$
$$\dot{m} = \alpha_m(V)(1-m) - \beta_m(V)m \tag{7.7}$$
$$\dot{h} = \alpha_h(V)(1-h) - \beta_h(V)h \tag{7.8}$$

where

$$\alpha_n(V) = \frac{(-0.01(V(t)+34))}{(\exp(-0.1(V+34))-1)} \tag{7.9}$$
$$\beta_n(V) = 0.071025\exp(-\frac{(V(t)+75)}{500}) \tag{7.10}$$
$$\alpha_m(V) = \frac{(-0.1(V+35))}{(\exp(-0.1(V+35))-1)} \tag{7.11}$$
$$\beta_m(V) = 4\exp(-\frac{V+68}{18}) \tag{7.12}$$
$$\alpha_h(V) = 0.07\exp(-\frac{(V+66)}{20}) \tag{7.13}$$
$$\beta_h(V) = \frac{1}{\exp(-0.1(V+28))+1} \tag{7.14}$$

To facilitate reading, units are dropped; voltages are in mV, time is in ms. Parameters are adapted from experimental values to yield a strong hyperpolarisation.

7.2.2 Learning Rule

The plasticity rule is derived from the demand of a balanced membrane potential: The neuron should neither be hyperpolarized nor too strongly depolarized. This is a sensible demand, because it holds the neuron at a sensitive working point and keeps metabolic costs down. To that end, two thresholds are introduced, $\vartheta_P < \vartheta_D < V_{thr}$, between which the membrane potential is bounded. The weight change is chosen such that, whenever $\vartheta_D = 10mV$ is surpassed, all weights that contribute to the rise of the membrane potential are depressed, weighted by their respective influence given by the PSP-kernel ε. Whenever the membrane potential drops below $\vartheta_P = V_L$, all synapses that contribute to that downward deflection are potentiated, such that for a repetition of the pattern the membrane potential is deflected to stay within bounds. Additionally, in some cases ($a = 1$) the weights are bounded to stay below a maximum weight w_{max}, symbolizing a maximal synaptic strength, while in other cases the weights are not limited ($a = 0$). Limiting the maximum weights is advantageous for stability in the case of synapses that can change signs and a real teacher input. The weight change is then given by

$$\dot{w}_i = \eta\left(w_{max} - |w_i|\right)^a \left(-\gamma\left[V(t) - \vartheta_D\right]_+ + \left[\vartheta_P - V(t)\right]_+^b\right) \sum_k \varepsilon\left(t - t_i^k - t_{delay}\right) \ . \tag{7.15}$$

The parameter b defines, whether the contribution of the potentiation term scales linearly ($b = 1$) or quadratically ($b = 2$) with the distance of the membrane potential from the plasticity threshold. γ is a factor that scales inhibition to excitation, which needs to be adapted to the neuron model in question. $\gamma = 650$ for the simple integrate-and-fire neuron

and $\gamma = 150$ for the conductance-based integrate-and-fire neuron. For the Hodgekin-Huxley type neuron, $\gamma = 20$.

Obviously, the PSP-kernel used in the learning rule only has a very direct interpretation in the case of simple linear integrate-and-fire neurons. However, in the learning rule, this term only serves to estimate the extent to which the postsynaptic membrane potential depends on the input of one particular neuron. Hence, in the non-linear neuron models a pseudo-PSP is tracked and used instead (with $\tau_m = C_m/g_L$). Furthermore, for the conductance-based integrate-and-fire neuron, the upper threshold is chosen to lie between resting potential and firing threshold: $\vartheta_D = -53mV$.
For the Hodgkin-Huxley type neuron, there is no spiking threshold. To avoid an overly strong influence of the very high membrane potential during the spike, the LTD part of the learning rule is clipped to the respective constant value at $\vartheta_D^{up} = -55mV$. The plasticity thresholds are chosen as $\vartheta_D = -58mV$ and $\vartheta_P = -64mV$.

For the inhibitory synapses coming into play for conductance-based neurons between the inhibitory presynaptic neurons and the output neurons, the learning rule is adapted such that the net effect of learning on the membrane potential is preserved. To that end, the same learning rule as for the excitatory synapses is applied, just with the opposite sign:

$$\dot{w}^{inh} = -\dot{w} \tag{7.16}$$

7.2.2.1 Chronotron Setup

Consider a feed-forward network consisting of $N = 200$ presynaptic neurons and one postsynaptic neuron. For illustration purposes, each input neuron spikes once in each of the five patterns used during training. To train the output neuron to spike at a specific time in response to each input pattern, a single spike is induced at a fixed time $t_{post} = 100ms$ by a supplementary external (teacher) input

$$I_{ext} = c \exp\left(-\frac{t - t_{post}}{\tau_s}\right) \Theta(t - t_{post}). \tag{7.17}$$

c is the amplitude of the teacher input. The shape of the current is chosen to mimic a synaptic input and induce a PSP-like voltage perturbance (see eq. (9.5)).

While in the case of the simple integrate-and-fire neuron synapses change signs, for the conductance-based integrate-and-fire neuron and the Hodgekin-Huxley type neuron positive and negative inputs need to be separated. To that end, the output neuron receives inhibitory input from $N_{in} = 200$ inhibitory presynaptic neurons and excitatory input from $N_{ex} = 200$ excitatory presynaptic neurons. An input pattern for learning thus consists of a set of one excitatory and one inhibitory input pattern. In each pattern, each input neuron spikes once.

During learning, the patterns are presented to the neuron concurrent with the teacher input. Weights are updated after all patterns were presented (batch mode). For recall, just the input patterns are presented and if learning was successful, the output neuron will generate a spike close to the time of the (now absent) teacher spike.

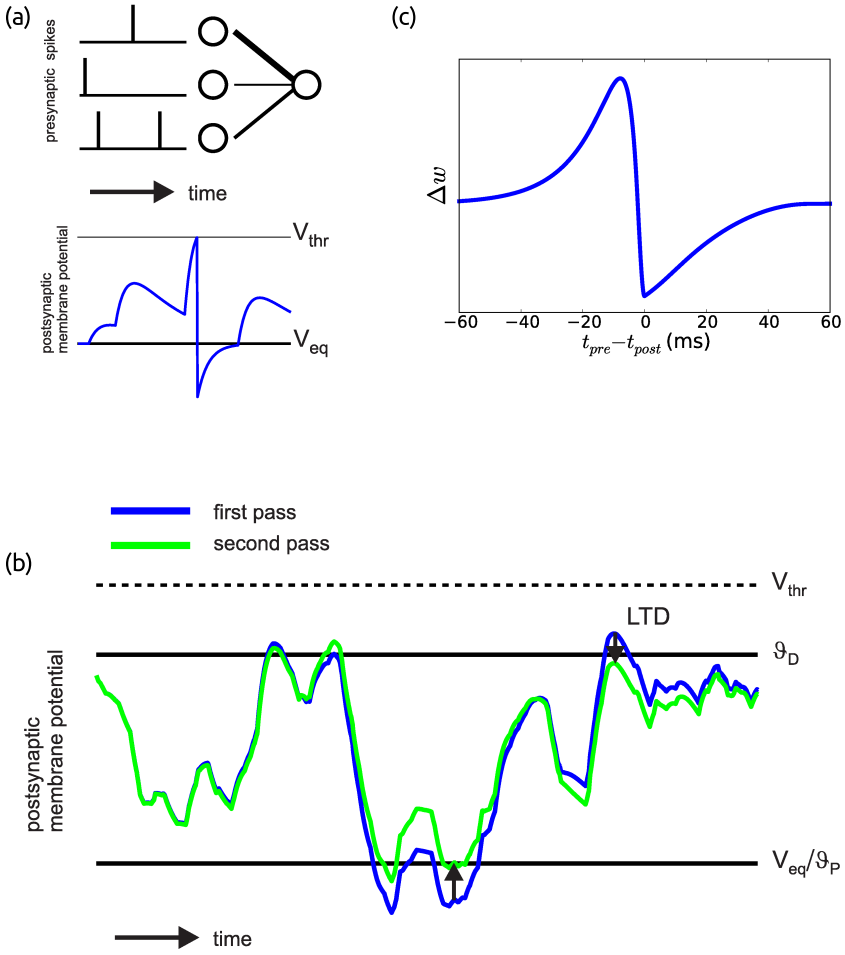


Figure 7.1: (a) The model network has a simple feed-forward structure. The top picture shows three pre- and one postsynaptic neurons, connected by synapses. Line width in this example corresponds to synaptic strength. The bottom picture shows the postsynaptic membrane potential in response to the input. (b) Illustration of Anti-Hebbian Membrane Potential Dependent Plasticity (MPDP). A LIF neuron is presented twice with the same presynaptic input pattern. Excitation never exceeds V_{thr}. MPDP changes synapses to counteract hyperpolarization and depolarization occuring in the first presentation (blue trace), reducing (arrows) them on the second presentation (green trace). (c) Homeostatic MPDP on inhibitory synapses is compatible with STDP as found in experiments. Plasticity is tested for different temporal distances between pre- and postsynaptic spiking; the resulting spike timing characteristic is in agreement with experimental data on STDP of inhibitory synapses [HNA06].

7.3 Results

7.3.1 Membrane Potential Dependent Plasticity

The learning algorithm is based on the interplay between a strong afterhyperpolarisation and a basic homeostatic requirement on the membrane potential of the neuron. The neuron should stay in a sensible working regime; in other words, its voltage should be confined to moderate values. This is formalized by introducing two thresholds on the voltage. ϑ_D lies between the firing threshold and resting potential and ϑ_P is equal to the resting potential. With these thresholds, an update rule for the weights is formulated (see equation 7.15). Weight changes with this rule "bend" the voltage at the times of non-zero error towards the region between the two thresholds. Figure 7.1 (b) shows how MPDP affects the voltage for recurring input activity.

7.3.2 Homeostatic MPDP on Inhibitory Synapses is compatible with STDP

First the biological plausibility of a network with MPDP is investigated. Experimental studies on plasticity of cortical excitatory neurons often find Hebbian plasticity rules like Hebbian Spike Timing Dependent Plasticity (STDP; see [MLFS97, Fel00, STN01, FD02, WGNB05] for examples). Reports on Anti-Hebbian plasticity or sensitivity to subthreshold voltage in excitatory cortical neurons are scarce [STN04, LHS$^+$07, FDV09, VGO$^+$13]. However, it has been reported that plasticity in (certain) inhibitory synapses onto excitatory cells has a Hebbian characteristic [HNA06], i.e. synapses active before a postsynaptic spike become stronger, those active after the spike become weaker. The net effect of this rule on the postsynaptic neuron is *Anti-Hebbian*, because weight increases tend to suppress output spikes.
In experimental investigations of STDP, neurons are tested with pairs of pre- and postsynaptic spikes. This procedure is mimicked in a simple network consisting of one pre- and one postsynaptic neuron, and one "experimentator neuron". The postsynaptic neuron is modelled as a conductance based LIF neuron. The experimentator neuron has a fixed strong excitatory synaptic weight onto the postsynaptic neuron, so that a spike of the experimentator neuron causes a postsynaptic spike. It serves to control the postsynaptic spike times. The presynaptic neuron is inhibitory and its weight is small compared to the experimentator, so that it has negligible influence on the postsynaptic spike time. Synaptic plasticity is probed by inducing a pair of a pre- and a postsynaptic spike at times t_{pre} and t_{post}, and varying t_{pre} while keeping t_{post} fixed. The resulting weight change of the inhibitory neuron as a function of timing difference is shown in figure 7.1 (c). The shape of the function is in qualitative agreement with experimental results [HNA06].
It is necessary to assume the presence of an "experimentator neuron". The reason for this is that the shape of the STDP curve explicitely depends on the specifics of spike induction since the MPDP rule is sensitive only to subthreshold voltage. For example, using a delta-shaped input current would lead to a LTD-only STDP curve, since the time the voltage needs to cross the firing threshold starting from equilibrium is infinitely short. These results were obtained for a linear contribution of the LTP part of the learning rule ($b = 1$), but qualitative shape of the STDP curve is similar for $b = 2$.

7.3.3 Homeostatic MPDP allows Associative Learning

At first glance, it might seem unlikely that a homeostatic plasticity mechanism can implement associative learning. It is Anti-Hebbian in nature, because if the membrane potential is close to the firing threshold it gets suppressed, and if it is below the resting potential it gets lifted up. However, the neuronal dynamics show somewhat stereotypic behaviour before, during and after each spike. To induce a spike, the neuron needs to be depolarized up to V_{thr}, where active feed-back processes kick in. These processes cause a very short and strong depolarization and a subsequent undershoot of the membrane potential (hyperpolarization), from where it relaxes back to equilibrium.

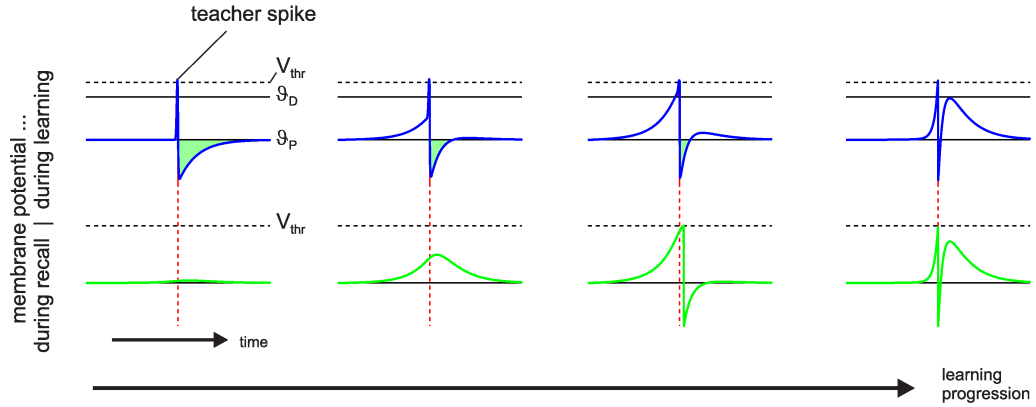


Figure 7.2: Hebbian learning with homeostatic MPDP. A postsynaptic neuron is presented the same input pattern multiple times, alternating between teaching trials with teacher spike (blue trace) and recall trials (green trace) to test the output. Initially, all weights are zero (left). Learning is Hebbian initially until strong depolarization occurs (second to left). When the spike first appears during recall, it is still not at the exact location of the teacher spike (second to right). Continued learning moves it closer to the desired location. Also, the time windows of the voltage being above ϑ_D and below ϑ_P shrink and move closer in time (right). Synaptic plasticity almost stops. The number of learning trials before each state is 1, 16, 53, and 1600 from left to right.

In a simplified setup, the basic learning mechanism is demonstrated. All synapses are subject to MPDP and are allowed to change their sign. A population of N presynaptic neurons fires one spike in each neuron at equidistant times. They project onto a single postsynaptic LIF neuron and all weights are zero initially. In each training trial an external delta-shaped suprathreshold current is induced at the postsynaptic neuron at a fixed time relative to the onset of the input pattern (teacher spike). The postsynaptic neuron reaches its firing threshold instantaneously, spikes and undergoes reset into a hyperpolarized state (blue trace on the left in figure 7.2). This is mathematically equivalent to adding a reset kernel at the time of the external current [MRÖS14]. Because $\vartheta_P = V_{eq} = 0$, potentiation is induced in all synapses which have temporal overlap of their PSP-kernel with the hyperpolarization. Probing the neuron a second time without the external spike shows a

small bump in the membrane potential around the time of the teacher spike. The same input pattern is repeated, alternating between teaching trials (with teacher spike) and recall trials without teacher and with synaptic plasticity switched off. Plasticity is Hebbian until the weights are strong enough such that there is a considerable depolarization before the teacher spike, inducing synaptic depression. Also, spike after-hyperpolarization is partially compensated by excitation, which reduces the window for potentiation. Continuation of learning after the spike association has been achieved (second to right plot) shrinks the windows for depression and potentiation, until they are very narrow and very close to each other in time. Because synaptic plasticity is determined by the integral over the normalized PSP during periods of depolarization and hyperpolarization, depression and potentiation become very similar in magnitude for each synapse and synaptic plasticity slows down nearly to a stop. Furthermore, the output spike has become stable. The time course of the membrane potential during teaching and recall trials is almost the same (figure 7.2 right).

7.3.4 Associative Learning with a real Teacher

To extend learning to a setup including a real teacher input of the shape of a regular synaptic input, first, the simplified setup is used in a simple integrate-and-fire neuron. Here, a limitation of the maximum weight is introduced to facilitate stability ($a = 1$). Furthermore, here, the contribution to the LTP part of the learning rule is quadratic, i.e. $b = 2$. This also facilitates stability in this simple neuron model, since the LTP contribution to the weight change around the spike decreases more quickly than the LTD contribution. Weights are initialized as zero. During the learning trials, a teacher induces a spike and consequently a strong hyperpolarisation (see figure 7.3.4, green lines). Before learning, the recall membrane potential is flat (red line, top). During the learning process, extra input is generated close to the teacher input, which fills up the hyperpolarisation. This extra input first just generates a bump in the recall membrane potential (second to top), until it generates a recall spike at a small distance to the teacher spike (middle row). The recall spike continues to become closer and closer to the recall spike, until learning settles and the spikes almost coincide (bottom).

7.3.5 Associative Learning in the conductance-based Integrate-and-Fire Neuron

To demonstrate the capability of MPDP for learning of exact spike times in more complex setups, the input population is split into N_i inhibitory and N_e excitatory neurons. Synaptic weights were initialized randomly. The splitting of neuron populations into excitatory and inhibitory neurons with just one set of synapses being plastic, in effect limits the range of the membrane potential. This renders a formal limit of the weights superfluous ($a = 0$). Here, the linear contribution of the LTP part of the learning rule was considered ($b = 1$). Both input populations project onto one conductance based LIF neuron. This network is presented with frozen poissonian noise as the sole presynaptic firing pattern (figure 7.4, top). Excitatory synapses were kept fixed and inhibitory synapses changed according to MPDP. First, the network learns to balance all inputs from the excitatory population such that the membrane potential mostly stays between the thresholds ϑ_P^I and ϑ_D^I. Then the teacher input is introduced as a strong synaptic input from a different source (e.g. a

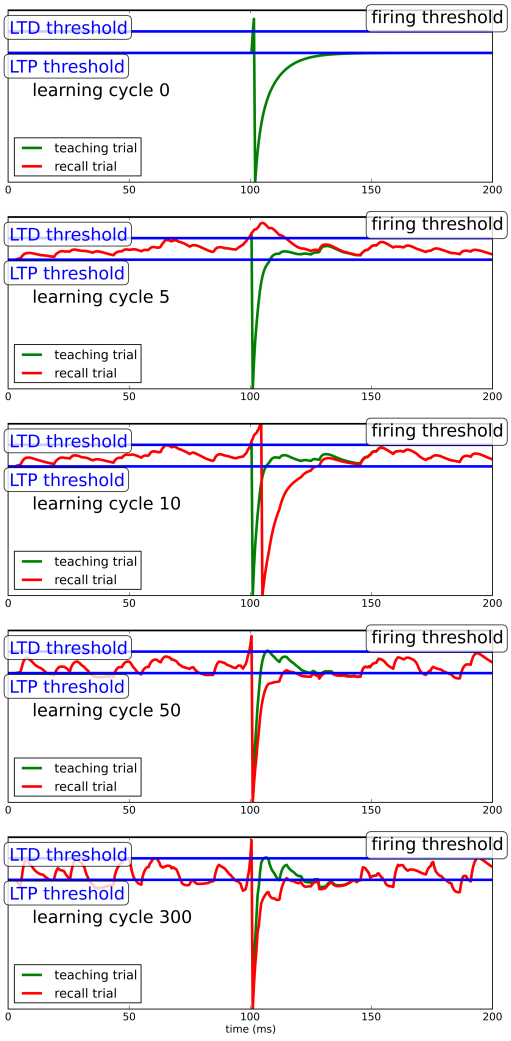


Figure 7.3: Learning progress in Chronotron toy model in the simple integrate-and-fire neuron. During teaching (green line), a regular, but strong input evokes a spike at the desired spike time. Due to the hyperpolarization after the teacher spike, the neuron adapts its synapses to generate extra input around the spike, which produces a spike when the teacher input is omitted (recall trial, red line). The learning progress is shown for several different stages of learning. Precise spike times can be learned in response to several input patterns independent of the specifics of the model neuron.

different neuron population, figure 7.4, second to top). After repeated presentations of the input pattern with the teacher input, inhibition around the teacher spike is released such that after learning the output neuron will spike close to the desired spike time even without the teacher input (figure 7.4, third and fourth to top). At the same time, due to the balance requirement of the learning rule, inhibitory and excitatory conductances covary and thus their influence on the membrane potential mostly cancels out (figure 7.4 bottom). Due to the stereotypical shape of the membrane potential around the teacher spike, a homeostatic learning rule is able to perform associative learning by release of inhibition.

7.3.6 Associative Learning in a Hodgekin-Huxley-type Neuron

To show that learning works in even more complex and realistic neuron models, MPDP is applied to the Hodgkin-Huxley type model.
Learning to balance the weights before the first teacher trial is dropped. Weights are initialized as zero and bounded. The quadratic component in the learning rule is used ($b = 2$).
Before learning, the teacher input elicits a spike, which drives the neuron into hyperpolarisation. After repeated presentation of pattern and teacher input, there is a small bump in the membrane potential upon recall (7.3.6, second to top). After learning progresses, a recall spike appears at some distance after the teacher spike, which continues to shift closer to the teacher as learning continues. After 500 learning steps, the recall spike almost coincides with the spike induced by the teacher input (bottom).

7.3.7 Other Results on MPDP

In our study [AWP16], we presented a quantitative assessment of the capacity of a simplified version of MPDP for unbounded weights ($a = 0$, see equation (7.15)) and a linear contribution of the LTP part of the learning rule in simple integrate-and-fire neurons. MPDP has about half of the maximum capacity of the Chronotron, which was theoretically estimated in [MRÖS14]. Furthermore, MPDP is much more robust to noise than all other learning rules due to the imposed distance of the membrane potential from the spiking threshold. Tolerance to noise comes at the cost of capacity. Since the quantitative results were obtained by Christian Albers, I will here just mention and discuss them. For these results and the details of the model see [AWP16].

7.4 Discussion

A synaptic plasticity mechanism was introduced that is based on the requirement to balance the membrane potential and therefore uses the postsynaptic membrane potential rather than postsynaptic spike times as the relevant signal for synaptic changes (Membrane Potential Dependent Plasticity, MPDP). It was shown that this simple rule allows the somewhat paradoxical temporal association of enforced output spikes with arbitrary frozen noise input spike patterns (Chronotron). Before, this task could only be achieved with

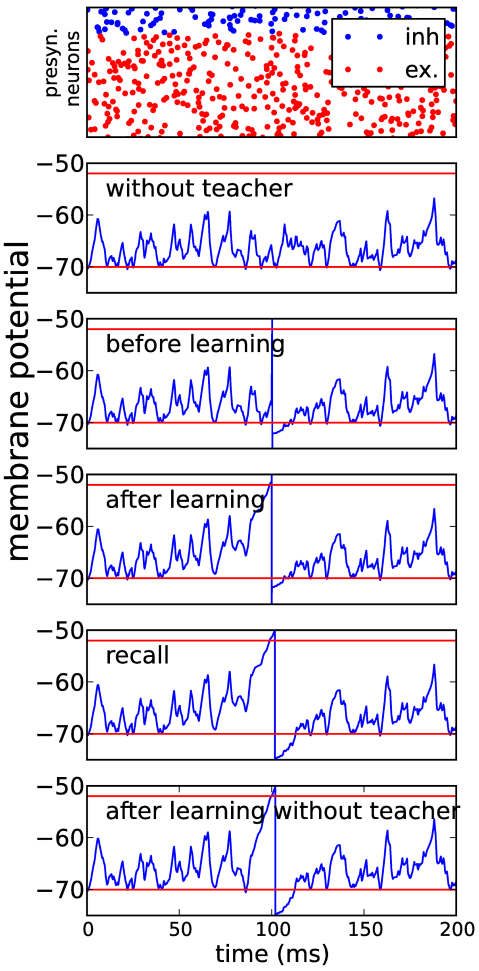


Figure 7.4: A conductance based integrate-and-fire neuron is repeatedly presented with a fixed input pattern of activity in presynaptic inhibitory or excitatory neuron populations (top row - blue is excitatory, red inhibitory). Before learning, the neuron is allowed to adapt its inhibitory weights according to homeostatic MPDP, such that the membrane potential mostly stays between the two learning thresholds. Then a strong excitatory input is given concurrently with the pattern to induce a spike at $t = 100ms$ (second row). Learning is restricted to inhibitory weights. By release of inhibition, the net input after the teacher spike is increased (third row). After learning has converged, the neuron is presented the input pattern without the teacher input and reproduces the spike close to the target time (4th row) . At all other times, excitatory and inhibitory conductances are balanced (bottom row).

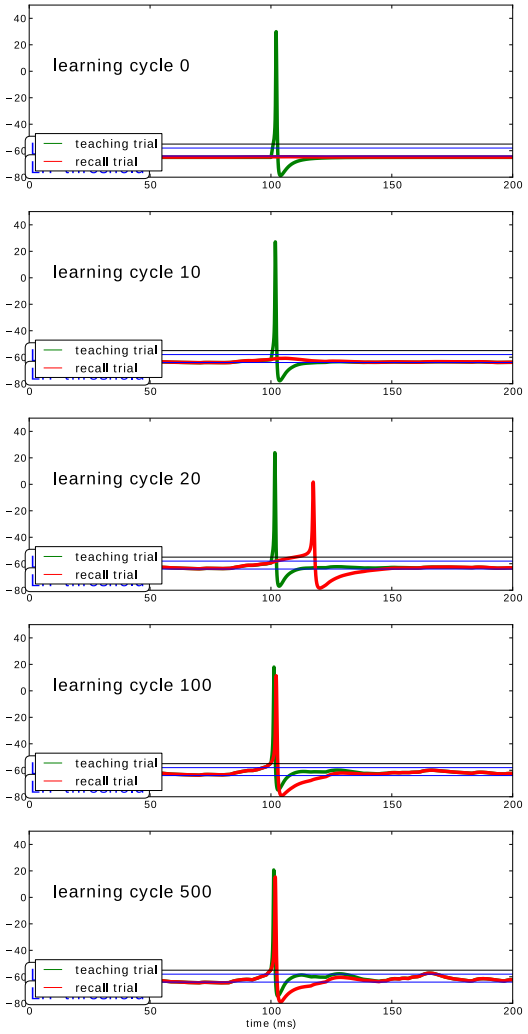


Figure 7.5: Associative learning in a Hodgkin-Huxley type neuron. Weights are initialized as zero, so before learning, the recall membrane potential is flat (red line, top; green line is the membrane potential during a teacher trial). After the input pattern is presented concurrently with a teacher input during some learning epochs, a small bump in the recall membrane potential appears (second row), which grows to elicit a spike at a certain distance from the teacher spike (middle row). This recall spike continues to shift to spike times closer and closer to the spike elicited by the teacher input, until it is almost identical (bottom line).

supervised learning rules that provided knowledge not only about the desired spike times, but also about the type of each postsynaptic spike (desired or spurious). With MPDP, the supervisor only has to provide the desired spike, while the synapse endowed with MPDP distinguishes between desired and spurios spikes exploiting the time course of the voltage around the spike. Additionally, the sensitivity of MPDP to subthreshold membrane potential allows for robustness against noise.

7.4.1 Biological Plausibility of MPDP

Spike-Timing-Dependent Plasticity (STDP) is experimentally well established and simple to formalize, which made it a widely used plasticity mechanism in modelling. It is therefore important to note that MPDP is compatible with experimental results on STDP, in particular with those of Hebbian STDP on inhibitory synapses. The reason is that spikes come with a stereotypic trace in the membrane potential. The voltage rises to the threshold, the spike itself is a short and strong depolarization, and afterwards the neuron undergoes reset, all of which are signals for MPDP. Pairing a postsynaptic spike with presynaptic spikes at different timings gives rise to a plasticity window which shares its main features with the STDP window: The magnitude of weight change drops with the temporal distance between both spikes and the sign switches close to concurrent spiking. It is known that the somatic membrane potential plays a role in synaptic plasticity. Many studies investigated the effect of prolonged voltage deflections by clamping the voltage for an extended time while repeatedly exciting presynaptic neurons (see e.g. [ABS90]). However, MPDP predicts that synaptic plasticity is sensitive to the exact time course of the membrane potential, as well as the timing of presynaptic spikes. This necessitates that dendrites and spines reproduce the time course of somatic voltage without substantial attenuation. Morphologically the dendritic spines form a compartement separated from the dendrite, which, for example, keeps calcium localized in the spine. It has been a topic under investigation whether the spine neck dampens invading currents. Despite experimental difficulties in measuring spine voltage, recent studies found that backpropagating action potentials indeed invade spines almost unhindered [HZK10]. Furthermore, independently of spine morphology and proximity to soma, the time course of a somatic hyperpolarizing current step is well reproduced in dendrites [PS09] and spines [PGCZ14]. This shows that at least in principle the somatic voltage trace can be available at the synapse. In turn, voltage-dependent calcium channels can transform subthreshold voltage deflections into an influx of calcium, the major messenger for synaptic plasticity. A few studies found that short depolarization events act as signals for synaptic plasticity [STN04, FDV09], with a dependence of sign and magnitude of weight change on the timing of presynaptic spikes. Another important point is the sign of synaptic change. "Membrane Potential Dependent Plasticity" per se is a very general term which potentially could include many different rules [CBVG10, SBC02]. Here, MPDP serves as a mechanism that keeps the membrane potential bounded. For inhibitory synapses this requirement results in a Hebbian plasticity rule, which has been reported previously [HNA06]. Inhibitory neurons in cortex have been implied to precisely balance excitatory inputs [XAS14]. MPDP on excitatory synapses is necessarily "Anti-Hebbian". Lamsa and colleagues [LHS$^+$07] found that pairing presynaptic spikes with postsynaptic hyperpolarization can lead to synaptic potentiation. This was caused by calcium permeable AMPA receptors (CP-AMPARs) present in these synapses. However, Anti-Hebbian plasticity does not rely on CP-AMPARs alone.

Verhoog et al. [VGO+13], for example, found Anti-Hebbian STDP in human cortex, which depends on dendritic voltage-dependent calcium channels. Taken together, these findings demonstrate the existence of cellular machinery which could implement homeostatic MPDP, either on excitatory or inhibitory synapses.

In the case of a simple integrate-and-fire neuron, the hyperpolarisation needs to be very pronounced, if a real teacher is used. This serves to eliminate the extra input from the teacher after the spike and to provide a strong learning signal, which is at odds with biological realism. It was shown that in a more sophisticated neuron model, such as a conductance-based integrate-and-fire neuron or the Hodgkin-Huxley type neuron, this overly strong hyperpolarisation is not necessary, because a sufficiently strong hyperpolarisation that is filled non-linearly during learning can be provided by an additional conductance.

7.4.2 Properties and Capabilities of Homeostatic MPDP

Homeostatic MPDP was derived from a balance requirement: Synapses change in order to prevent hyperpolarization and strong depolarization for recurring input activity. This kind of balance reduces metabolic costs of a neuron and keeps it at a sensible and sensitive point of operation [AL01]. The resulting plasticity rule is Anti-Hebbian in nature because synapses change to decrease net input when the postsynaptic neuron is excited and to increase net input when it is inhibited. However, spike after-hyperpolarization turns homeostatic MPDP effectively into Hebbian plasticity. Every postsynaptic spike causes a voltage reset into a hyperpolarized state. Therefore synapses of presynaptic neurons which fired close in time to the postsynaptic spike will change to increase net input if the same spatio-temporal input pattern re-occurs. The total change summed over all synapses depends on the duration and magnitude of hyperpolarization. Because the induced synaptic change reduces this duration, total synaptic change is also reduced. The same is true for total synaptic change to decrease net input, which depends on the duration of the membrane potential staying above ϑ_D (resp. ϑ_P^I for inhibitory synapses) and which reduces this duration in future occurrences. If the rise time of the voltage before the spike and residual spike after-hyperpolarization are both short and close in time, potentiation and depression will become approximately cancelled around a spike.

In this view, associative synaptic plasticity or "learning" is the consequence of imbalance. A spike is stable if the time course of the voltage in its proximity leads to balanced weight changes. For example, if input is just sufficient to cause a spike, the voltage slope just before the spike is shallow and synaptic depression outweighs potentiation. On the other hand, the delta-pulse shaped currents used to excite the postsynaptic neuron during Chronotron training are very strong inputs. They are not unlearned. Instead, the weights potentiate until the membrane potential is in a balanced state, and the neuron fires the teacher spike on its own when left alone.

Another interesting aspect of MPDP is the emergence of robustness against noise. Most obviously, with the choice of the threshold for depression the neuron sets a minimal distance of the voltage to the firing threshold for known input patterns. This allows to have perfect recall in the case of noisy input in the Chronotron. The second effect of the depression threshold is more subtle. Not only does it prevent spurious spikes, but through learning the slope of the membrane potential just before the desired spike tends to become steep. This is necessary to prevent spike extinction by noise. To see how this influences

noise robustness, consider an output spike with a flat slope of the voltage. Increasing the voltage slightly around the spike time moves the intersection of the voltage with the firing threshold forward in time by a proportionally large margin. Decreasing voltage moves it backwards in time or could even extinguish the spike; a flat slope implies a low peak of the "virtual" membrane potential. MPDP in contrast achieves a state which is robust against spike extinction as well as the generation of spurious spikes. On the downside, keeping the voltage away from the firing threshold as well as imposing steepness on the slope just before spikes puts additional constraints on the weights. Robustness comes at the cost of capacity.

7.4.3 Relation of MPDP to other Learning Rules

There are many supervised learning algorithms that are used to train neuronal networks to generate desired spatio-temporal activity patterns. All of them involve a comparison of the self-generated output to the desired target activity. They can be broadly put into three different classes. E-Learning and FP-Learning [Flo12a, MRÖS14] are examples of algorithms of the first class which are used to train a neuron to generate spikes at exactly defined times. They first observe the complete output and then evaluate it against the target. E-Learning performs a gradient descent on the Victor-Purpura distance [VP96] between both spike trains. This means that the weight changes associated to one particular spike (actual or desired) can depend on distant output spikes. In FP-Learning, the training trial is interrupted if the algorithm encounters an output error. Subsequent spikes are not evaluated anymore. Thereby these algorithms are non-local in time and very artificial. Another class of learning algorithms emerged recently with the examples of PBSNLR [XZZ13] and HTP [MRÖS14]. They take an entirely different route. The postsynaptic membrane potential is treated as a static sum of PSP kernels weighted by the respective synaptic weight, similar to the SRM_0 model of the LIF neuron. The firing threshold is moved towards infinity to prevent output spikes and voltage resets are added at the target spike times. Then the algorithms perform a Perceptron classification on discretely sampled time points of the voltage, with the aim of keeping it below the actual firing threshold for all non-spike times and to ensure a threshold crossing at the desired spike times. These algorithms were devised as purely technical solutions and are highly artificial. However, MPDP bears some similarity to the described procedure: Except close to teacher inputs, at every point in time recently active synapses get depressed if the voltage is above the threshold for depression. This is comparable to a Perceptron classification on a continuous set of points.
A third class of algorithms compares actual and target activity locally in time. In contrast to the algorithms mentioned above, they are usually not used to learn exact spike times, but rather continuous time dependent firing rates. The ur-example is the Widrow-Hoff rule [HKP91, PK10]. More recently, similar rules were developed by Xie and Seung [XS04], Brea et al. [BSP13] and Urbanzcik and Senn [US14]. In contrast to the Widrow-Hoff rule, the more recent rules are defined for spiking LIF neurons with a "soft" firing threshold, i.e. spike generation is stochastic and the probability of firing a spike is a monotonous function of the current voltage. In these rules, at every point in time the synaptic change is proportional to the difference of the current firing rate and a target firing rate specified by an external supervisor. When it comes to biological implementation, the central problem of Widrow-Hoff type rules is the comparison of self-generated and target activity. It

is derived from the abstract goal to imprint the target activity onto the network. This target needs to be communicated to the neuron and synaptic plasticity has to be sensitive to the difference of the neurons' own current activity state (implicitly represented by its membrane potential) and the desired target activity. Usually, no plausible biological implementation for this comparison is given. The combination of homeostatic MPDP, hyperpolarization and a teacher now offers a solution to both problems. The teacher provides information about the target activity through temporally confined, strong input currents which cause a spike. Spike after-hyperpolarization (SAHP) allows to compare the actual input to the target without inducing spurious spikes detrimental to learning. The more SAHP is compensated by synaptic inputs, the closer the self-generated activity is to the target and the less synapses need to be potentiated. This is implemented naturally in MPDP, where potentiation is proportional to the magnitude and duration of hyperpolarization. On the other hand, strong subthreshold depolarization implies that self-generated spurious spikes are highly probable, and weights need to be depressed to prevent spurious spikes in future presentations.
A further solution for the problem of how information about the target is provided was given by Urbanczik and Senn [US14]. Here, the neuron is modelled with soma and dendrite as seperate compartements instead of point neurons as used in this model. The teacher is emulated by synaptic input projecting directly onto the soma, which causes a specific time course of the somatic membrane potential. The voltage in the dendrite is determined by a different set of synaptic inputs, but not influenced by the somatic voltage; however, the soma gets input from the dendrites. The weight change rule then acts to minimize the difference of somatic (teacher) spiking and the activity as it would be caused by the current dendritic voltage. This model represents a natural way to introduce an otherwise abstract teacher into the neuron. Nonetheless, the neuron still has to estimate a firing rate from its current dendritic voltage, for which no explicit synaptic mechanism is provided. Also, it is worth noting that the model of Urbanczik and Senn requires a one-way barrier to prevent somatic voltage invading the dendrites; in contrast, MPDP requires a strong two-way coupling between somatic and dendritic/synaptic voltage.
Another putative mechanism for a biolgical implementation of the δ-rule was provided by D'Souza et al. [DLH10]. In this model, a neuron recieves early auditory and late visual input. By the combination of spike frequency adaptation (SFA) and STDP, the visual input acts as the teacher that imprints the desired response to a given auditory input in an associative manner. However, the model is quite specific to the barn owl setting; for example, parameters have to be tuned to the delay between auditory and visual input.
Applying rules of the Widrow-Hoff type to fully deterministic neurons can lead to unsatisfactory results. ReSuMe is an example of such a rule [PK10]. Its memory capacity is low, but it increases sharply if the input is noisy during training (see [AWP16] for details). A propable reason is that in a fully deterministic setting, the actual spike times do not allow a good estimation of the expected activity. This sounds paradoxical. But if we consider a deterministic neuron with noise-free inputs, the membrane potential can be arbitrarily close to the firing threshold without crossing it. But even the slightest perturbation can cause spurious spikes at those times. This leads to bad convergence in Chronotron training, since the perturbations caused by weight changes for one pattern can easily destroy previously learned correct output for another pattern [MRÖS14]. The problem of these rules is the sensing of the activity via the instantaneous firing rate. Therefore, the explicit sensitivity to subthreshold voltages of MPDP is advantageous if training examples are

noise free.
In conclusion, the MPDP rule with hyperpolarization and teacher input represents a biologically plausible implementation of the comparison of actual and target activity that is key to all supervised learning algorithms. Also, because MPDP is explicitely sensitive to the membrane potential and not the firing rate, it is fully applicable to deterministic neurons. Additionally, the training procedure leads to networks whose output is robust against input noise, similar to what learning algorithms of the Widrow-Hoff type achieve.

Chapter 8

Learning of Inverse Models with Membrane Potential Dependent Plasticity

In this chapter, a different setup for the learning of inverse models will be introduced. For maximal comparability with chapter 5, the model of the interaction of the learner with the outside world and some of the evaluated quantities are chosen the same. However, for the sake of completeness I will repeat the descriptions here.

This model was conceived to remedy the shortcomings of the simple model introduced in chapter 5. In that model, it was found that the inversion of a world mapping that maps individual motor spikes onto too long sequences of sensory patterns is difficult. Here these elongated patterns are mapped back onto simpler, shorter ones - which is equivalent to the Chronotron problem.

8.1 Introduction

Inverse sensor-motor models serve to generate a desired sensory input by appropriate motor actions. In this sense they attempt to 'invert' the action-sensation mapping given by the physical world. While in general this mapping is not stationary, sound sequence imitation represents a comparatively well controlled situation. Therefore, it was tempting to propose inverse models as the mechanism enabling many bird species to imitate previously heard acoustic signals [HGH14]. The underlying hypothesis is that inverse models in the bird's brain perform a transformation of memorized sensory representations of sound sequences into spatio-temporal patterns of activities in motor areas that in turn generate the same sound sequences. This enables imitation of arbitrary sound sequences within the realm of the possible sounds the bird can produce. A crucial prediction of such so called causal inverse models is the existence of mirror neurons active during both singing as well as playback of a recording of the bird's song. The responses of these mirror neurons to a playback would be delayed relative to the bird itself singing the song. This delay reflects the loop time it takes for motor activations to produce sound, which produces sensory activations that are looped back to the respective motor area (e.g. about 40 ms in zebra finches). Indeed, a recent study has found evidence for such delayed mirroring in area

LMAN of the songbird [GKGH14] (for more information on auditory-vocal mirroring in songbirds see section 2.2.3).

The unambiguously clear mirroring with roughly zero delay discovered in neurons in area HVC_x of swamp sparrows [PPNM08], however, is at odds with previous explanations using causal inverse models. It was suggested to reflect a 'predictive inverse model' that could emerge from hebbian learning of a single stereotyped (i.e. predictable) song [HGH14]. If this was true, these neurons could not be (directly) involved in imitation of arbitrary sound sequences, i.e. their zero delay mirroring would represent a highly specific epiphenomenon emerging from a system enabling reproduction of a limited set of memorized sensory experiences of sounds.

Here, an alternative causal inverse model is proposed in which zero-delay mirroring rather reflects a delayed feedback from motor areas backwards to HVC that compensates for the loop delay. The architecture consists of three interacting neuronal modules that could be identified with corresponding areas in the songbird involved in sound production. In particular, it includes the hypothetical feedback which for conceptual simplicity is realized by delay-lines (see figure 8.1(a)).

The delayed feedback turns out to be particularly beneficial for solving the problem of learning inverse models when they are based on precise spatio-temporal spike patterns, because it can then be mapped to the problem of Chronotron learning (see chapter 7 and [Flo12b, MRÖS14, GS06, PK10]). The learning mechanism introduced in chapter 7 will therefore here be applied to the learning of inverse models.

This mechanism can not only learn simple Chronotrons (see chapter 7), but is similarly potent for learning the mappings of spatio-temporal spike patterns to spatio-temporal spike patterns as required in inverse models. In particular, it will be shown that zero-delay mirroring in the model presented here naturally emerges in HVC_x neurons that receive - either by chance or because of anatomical constraints - no direct sensory memory input.

8.2 The Model

8.2.1 Network Setup

To investigate learning of inverse models, a model of connected neuronal populations reflecting the brain anatomy of songbirds is constructed (see figure 8.1(a)).

A population of N_m motor neurons in the motor area m activates the muscles in the syrinx for singing. The bird's cochlea converts sounds into activations of N_s sensory neurons in population s. s innervates a sensorimotor area $s_x m_x$ reps. $s_2 m_2$ via plastic synapses. This sensorimotor area serves as a relay for forming the inverse model.

To mimick the 'babbling' young birds presumably use to establish the relation of motor activities with the corresponding sensations of self-generated sounds, the motor neurons are fed strong delta-shaped current pulses of height $h = 0.5$ with a frequency of r_{explor} during an exploration phase. The firing rate of the neurons in motor population m is limited by the long hyperpolarization in this area introduced by the long membrane time constant τ_m^m. This long hyperpolarization serves to suppress cyclic activity between areas m and $m_2 s_2$.

Consequently, the spatio-temporal motor activity is transformed into input into the

sensory neurons via the world model (see section 8.2.4), which in turn create spatio-temporal sensory spike patterns.

Note that the sensory area s is split into two sub-populations s_{sens} and s_{recall} receiving the same input. Only s_{recall} will be activated to retrieve the memory of the tutor song during recall, while s_{sens} only receives sensory input.

At all times, a copy of the motor activations in m is send as teacher input to a population of sensorimotor neurons in s_2m_2 ($\tau_m^{sm} = 8ms$) with a delay τ_{loop} equivalent to the loop delay. The teacher input takes the shape of a synaptic input (see neuron model, section 8.2.2) of strength $c = 0.3$:

$$I_{ext} = c \exp\left(-\frac{t - t_{post}}{\tau_s}\right) \Theta(t - t_{post}). \tag{8.1}$$

By this teacher input the spatio-temporal spike patterns activated in sensory memory area s_{recall} can be mapped back onto the delayed copy of the motor patterns in the sensorimotor population s_2m_2. The sensorimotor population s_2m_2 then gives a copy of it's own activation as input into the motor population. The synaptic weights from the sensory population s to the sensorimotor population s_2m_2 are plastic according to (8.6).

8.2.2 Neuron Model

All neurons are modelled as simple leaky integrate-and-fire neurons. To facilitate the derivation of the plasticity rule, the formulation of the SRM_0 [GK02] is used. A neuron j receives input from other neurons i either via plastic synapses of weight w_{ji} or via the model of the sound generation and perception process (see section 8.2.4). The neuronal voltage $V(t)$ is given by the sum of weighted synaptic input kernels $\varepsilon(s)$ (postsynaptic potentials, PSPs) and reset kernels $R(s)$, which model the neuronal reset after a spike. External input currents $I_{ext}(t)$ are low-pass filtered with a response kernel $\kappa(s)$. The full equation reads

$$V_j(t) = \sum_i w_{ji} \sum_k \varepsilon(t - t_k^i - t_{delay}) + \sum_{t_j} R(t - t_j) + \int_0^\infty \kappa(t - s) I_{ext}(s) ds. \tag{8.2}$$

Here, w_{ji} is the weight from presynaptic neuron i to the postsynaptic neuron j and t_i^k is the time of the k-th spike of presynaptic neuron with index i. A delay of synaptic transmission t_{delay} is included. The synaptic input kernel, the reset kernel and the passive filtering kernel are given by

$$\varepsilon(s) = \Theta(s) \frac{1}{\tau_m - \tau_s} \left(\exp(-s/\tau_m) - \exp(-s/\tau_s)\right) \tag{8.3}$$

$$R(s) = \Theta(s)(V_{reset} - V_{thr}) \exp(-s/\tau_m) \tag{8.4}$$

$$\kappa = \exp\left(-(t - s)/\tau_m\right). \tag{8.5}$$

where τ_m is the membrane time constant of a LIF neuron determining the decay of voltage perturbations, and $\tau_s = 2ms$ is the decay time constant of synaptic currents, which defines the rise time of the PSP kernel. If there is no input, the voltage relaxes back to $V_{eq} = 0$. Spiking in this model is deterministic: If $V(t') = V_{thr} = 20mV$, the neuron spikes and a reset kernel is added at time $t' = t_j$. The formulation of the kernel makes sure that the voltage is always reset to $V_{reset} = -60mV < V_{eq}$. The membrane time constants of the neurons in the respective areas are $\tau_m^m = 70ms$ and $\tau_m^s = \tau_m^{sm} = 8ms$.

8.2.3 Learning Rule

The plasticity rule is derived from the demand of a balanced membrane potential: the neuron should neither be hyperpolarized nor too strongly depolarized. This is a sensible demand, because it holds the neuron at a sensitive working point and keeps metabolic costs down. To that end, two thresholds are introduced, $\vartheta_P < \vartheta_D < V_{thr}$, between which the membrane potential is bounded. The weight change is chosen such that, whenever $\vartheta_D = 10mV$ is surpassed, all weights that contribute to the rise of the membrane potential are depressed, weighted by their respective influence given by the PSP-kernel ε. Whenever the membrane potential drops below $\vartheta_P = V_L$, all synapses that contribute to the membrane potential at the time of that downward deflection are potentiated, such that for a repetition of the pattern the membrane potential is deflected to stay between bounds. Additionally, the weights are bounded to stay below a maximum weight $w_{max} = 1.5$, symbolizing a maximal synaptic strength. Limiting the maximum weights is advantageous for stability. The weight change is then given by

$$\dot{w}_i = \eta \left(w_{max} - |w_i|\right) \left(-\gamma \left[V(t) - \vartheta_D\right]_+ + \left[\vartheta_P - V(t)\right]_+^2\right) \sum_k \varepsilon \left(t - t_i^k - t_{delay}\right) \ . \quad (8.6)$$

$\gamma = 1000$ is a factor that scales inhibition to excitation. η is the learning rate.

8.2.4 World Model

To model the bird hearing its own vocalizations, spatio-temporal activity in m is converted to input in s through one or several delayed linear transformations. N_w defines the number of different delays and τ_w is the temporal difference of the delays relative to each other. τ_{ms} is the median delay. To construct this model of the world, a sparse matrix M_{all} is created, where each entry is either zero, a positive constant with probability $P_p = 0.1$ or a negative constant with probability $P_n = 0.1$. Then N_w empty matrices are constructed and the content of M_{all} is distributed over these matrices with equal probability by assigning each entry of M_{all} a delay out of the set of the N_w different delays associated with the N_w matrices M_r, such that finally $\sum_r M_r = M_{all}$. This construction of the world ensures maximum comparability between the most simple case of just one delay and the more complex ones.

To then generate input I_i into neuron i in the sensory population s, spikes in m are low-pass filtered by $\tau_s \dot{\vec{y}} = -\vec{y} + \vec{m}$:

$$I_i(t) = \sum_r^{N_w} M_r^i \vec{y}(t - \tau_{ms} - (r - \frac{1}{2} N_w)\tau_w). \quad (8.7)$$

This generates a nonlinear transformation of motor activities into sound activities, which may or may not be local in time.

8.2.5 Measuring the Learning Progress

8.2.5.1 General Measuring Procedure

To evaluate learning progress, one particular motor pattern with input rate r_{song} into the motor area during exploration and its respective sensory pattern are picked out and

assigned the role of the tutor song; they are stored for later comparison. This is done to ensure that the model bird can in principle generate the desired activity sequence and is equivalent to assuming the tutor bird and the student bird to have the same mapping from motor activity to sound output. The firing rate of exploration during learning r_{explor} can be the same or different from the song firing rate r_{song}. Learning and recall periods are of duration $T = 3000ms$. Weights are initialized as zero, the world model is initialized at random for every trial according to section 8.2.4.
To allow for learning, in the exploration phase, random input is given into the motor population m during $N_k = 2000$ learning epochs. During this learning phase, all weight changes are summed up and applied after each training period. Every $\Delta N_k = 50$ learning epochs, recall is tested.
To test the ability of the system to reproduce the tutor song, the sensory representation of the tutor song is set to be the activity in the sensory area s by hand (in addition to any self-generated activity). This sensory activity then generates some sensorimotor activity in area s_2m_2, which in turn generates motor activity in area m, which - if learning is successful - is a shifted version of the 'tutor' motor activity. This recall motor activity is in turn fed into a copy of the sensory population to test whether it generates the same sensory impression on the model bird that the tutor song did. Testing whether the same sensory impression is generated is equivalent to testing if the song sounds the same to the (model) bird, which is the marker of good imitation. It is in principle possible that two very different motor sequences generate the same sound output and therefore the same sensory impression. Since for the bird the emphasis is on mimicking the sound, this is the relevant measure of success. As we will see, however, the difference between learning success measured on the motor patterns and learning success measured on the sensory patterns is very small, if learning is successful. For maximum comparability between the learning trials, the learning rate is scaled with the firing rate and the system size to yield a similar change on the membrane potential per learning cycle, such that $\eta_{norm} = \eta N_s r_{explor}/N_w = 1 \cdot 10^{-5}$ is the same for all trials.

8.2.5.2 Measure of Pattern Similarity

To quantify the similarity of the tutor pattern and the self-generated pattern, it is necessary to compare two sets of spike trains. The activity $a_i^s(t)$ in each neuron i in the tutor song will have to be compared to the activity during recall $a_i^r(t)$ to give an appropriate distance measure $d_0(a_i^s(t), a_i^r(t))$. The total distance over the activity a^s resp. a^r of all neurons in the given population will then just be the sum over all neurons in the population. Finally, this distance should be minimized over a global shift to account for the loop delay and normalized to the number of spikes in the tutor pattern:

$$d(a^s, a^r) = \min_{\Delta t} \sum_i d_0(a_i^s(t), a_i^r(t - \Delta t)) \tag{8.8}$$

This quantity is evaluated every $\Delta N_k = 50$ learning cycles. The resulting learning curves are normalized to the number of spikes in the tutor pattern, such that the error before learning is 1. For small N_s, it is possible that the sensory tutor pattern does not contain any spikes. Since this is equivalent to a tutor song without sound, these trials are discarded and repeated with a different initialization.
For the quantitative analysis, the residual error after learning is computed by taking the

average over the last 10% of learning steps in each of the $N = 50$ trials and then computing average and standard error from those measurements.

8.2.5.3 Spike Train Distance Measures

There are several possible spike train distance measures $d_0(s_1, s_2)$, e.g. the VanRossum-distance [vR01] and the Victor-Purpura-distance [VP96].

To calculate the VanRossum-distance between two spike trains s_1 and s_2, both spike trains are convoluted with an exponential kernel. Then the quadratic distance is computed between those convolutions. While this spike train distance measure is easy to implement, it has the computational disadvantage of the computing time being dependent on the total number of simulation time steps.

Calculating the Victor-Purpura-distance seems more complicated, but is generally faster for not too high firing rates: To evaluate a spike train distance between spike trains s_1 and s_2, a cost for the transformation from s_1 into s_2 is calculated. There is a cost of 1 for the deletion or introduction of a spike and a cost of $q\Delta t$ for a shift of the spike time of one spike by Δt, where q is a parameter that scales the cost of shifting a spike relative to the insertion and deletion of spikes. The sum of the costs to transform s_1 into s_2 is then the spike train distance $d(s_1, s_2)$.

8.2.6 Autocorrelation Function

In the setup of the model, the membrane time constant of neurons in the motor population is very long, which leads to a imposed distance between spikes in these neurons. To be able to quantify this experimentally accessible property of the model, the autocorrelation function is introduced.

It serves to evaluate how the spiking probability of the motor neurons depends on past spiking activity and is given by $\rho(\Delta t)$:

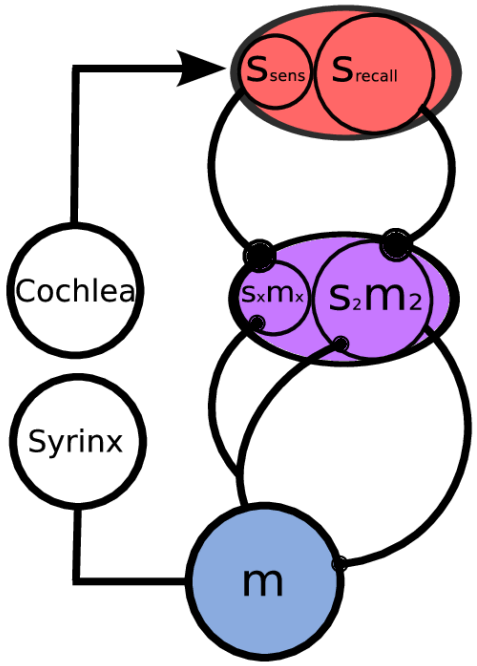
$$\rho(\Delta t) = \frac{\langle (m(t) - \bar{m})(m(t - \Delta t) - \bar{m}) \rangle_{N_m N}}{\sigma_m^2} \tag{8.9}$$

where $< \cdots >_{N_m N}$ denotes the average over motor neurons and the ensemble, $\bar{m}$ is the mean and σ_m is the standard deviation of $\vec{m}(t)$. In this form, the autocorrelation function is normalized between -1 and 1, 1 indicating perfect correlation, -1 indicating perfect anti-correlation. An autocorrelation of 0 indicates uncorrelated spiking activity.

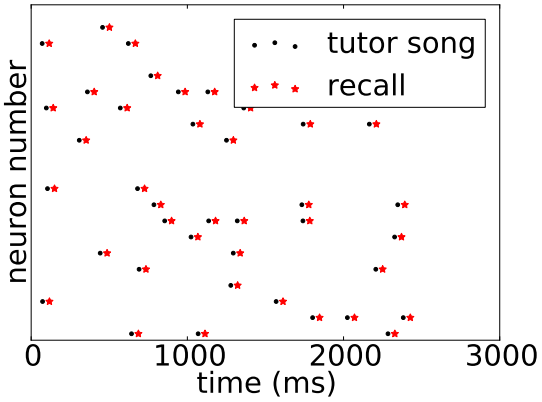
8.3 Results

8.3.1 Basic Learning Mechanism

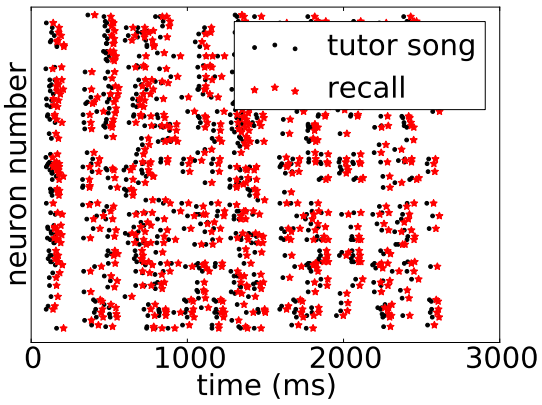
To investigate learning of inverse models, a model of connected neuronal populations reflecting the brain anatomy of songbirds is introduced (see figure 8.1(a)). A population m in the motor area activates the muscles in the syrinx for singing. The bird's cochlea converts sounds into activations of neurons in sensory area s. To model the bird hearing its own vocalizations, spatio-temporal activity in m is converted to input in s through several delayed linear transformations. This temporally spreads the self-generated sounds around the loop delay. Mimicking the 'babbling' young birds presumably use to establish



(a) Inverse model setup



(b) motor neurons



(c) sensory neurons

Figure 8.1: (a) Sketch of the system setup. (b),(c) Motor resp. sensory pattern. Black dots mark target spike times and red stars mark recall activity. For the sensory pattern, only half of the $N_s = 400$ sensory neurons are displayed.

the relation of motor activities with the corresponding sensations of self-generated sounds, the motor neurons are driven with noise during an exploration phase (see figure 8.1(b) for an example pattern). Consequently, the spatio-temporal motor activity is transformed into input into the sensory neurons, which in turn create spatio-temporal sensory spike patterns (see figure 8.1(c)).

A successful inverse model then has to map these sensory patterns, when retrieved from memory, back onto the relatively sparse motor patterns that have generated the respective sensory inputs, which is a task similar to the Chronotron task (see chapter 7). Note that the sensory area s is split into two sub-populations s_{sens} and s_{recall} receiving the same input given by (8.7). Only s_{recall} will be activated during recall, while s_{sens} only receives sensory input.

Before learning the inverse model, a memory of the tutor song is formed by choosing one particular random pattern in motor area m and the respective sensory pattern, which are stored for later comparison. Since the target pattern is chosen to be a particular stochastic pattern with the same or at least similar statistics as the training set, it could

by chance occur during the babbling phase. Due to the stochastic nature of the exploration, however, this is highly unlikely. This choice of target pattern is equivalent to assuming that the tutor bird has the same mapping from motor activity to sound and thus to auditory activity. This ensures that the resulting song (i.e. sensory activation) can in principle be generated perfectly by the model bird.

During learning, the stored motor pattern is compared to the motor pattern that is evoked when the tutor sensory pattern is fed into the sensory population (recall case). This motor pattern is then used to test which sensory pattern it would evoke. Figure 8.1(b) and 8.1(c) show spike raster plots of the target motor resp. sensory activity (black dots) and recalled activity via the inverse model (red stars) for $r_{target} = r_{explor} = 1Hz$. After learning, the tutor pattern is very well reproduced in both the motor and the sensory area with a time delay of about τ_{loop}.

8.3.2 Quantitative Evaluation of the Learning Process

For the quantitative evaluation of the learning process, the spike train distance between patterns of motor resp. sensory activity is measured every $\Delta N = 50$ learning epochs over a total of $N_k = 2000$ learning epochs. A typical set of learning curves, averaged over $N = 50$ sets of initializations, for an exploration and song firing rate $r_{explor} = r_{test} = 1Hz$ is displayed in figure 8.2. Learning is quick and after learning, the system settles at a low error. When learning is successful, learning curves for the motor pattern and the sensory pattern are similar. Note that in case of a relatively high residual error ($N_m = 10$, purple line), the error is slightly higher in the motor population than in the sensory population. This is a trace of the fact that different motor patterns do not give sufficiently different sensory patterns, such that the mapping is difficult to invert.
For all further evaluations, a residual error is calculated by taking the average of the last 10% of learning steps for each initialization. From this set of residual error measurements, average and standard error are computed and used in all further investigations.

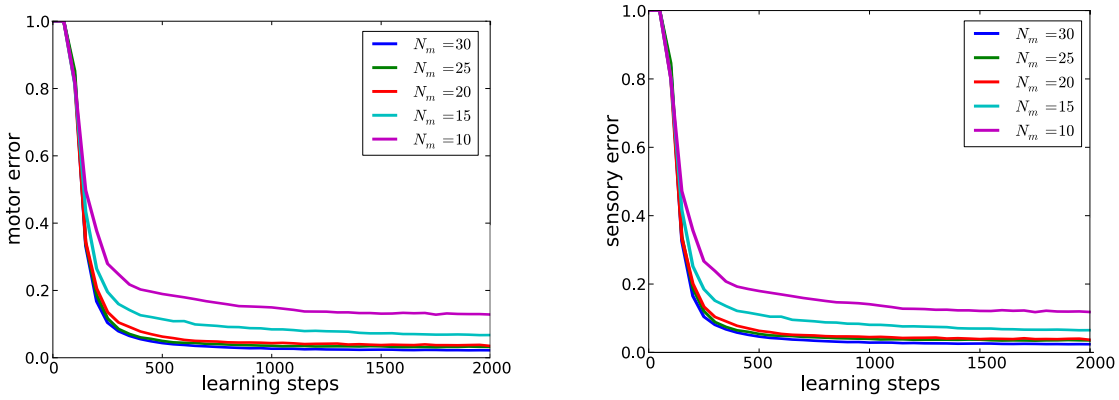


Figure 8.2: Learning curves for $N_m = \{10, 15, 20, 25, 30\}$ and $\alpha = 20$ in a complex world model with $N_w = 40$. Over learning, the error decreases quickly and settles on a low level. The residual error is calculated by taking the average over the last 10% of learning steps for individual learning curves. From these residual errors average and standard error are calculated. Left: motor population, right: sensory population.

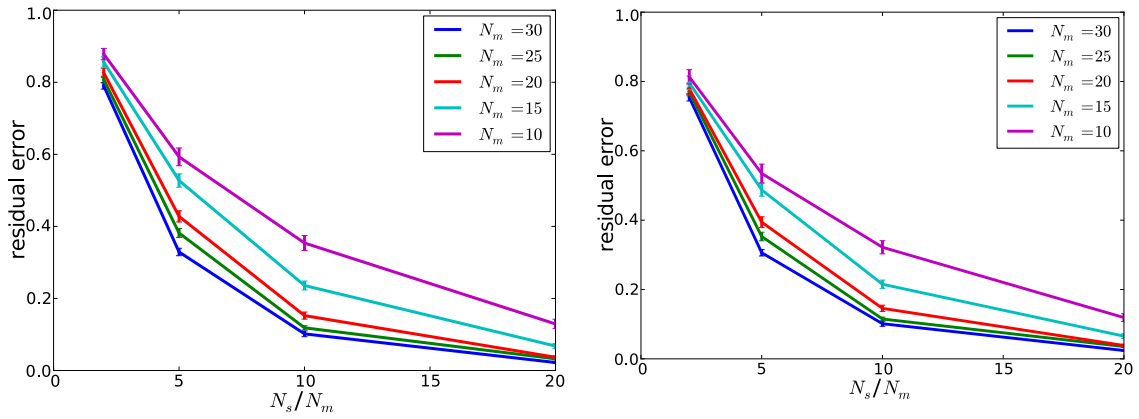
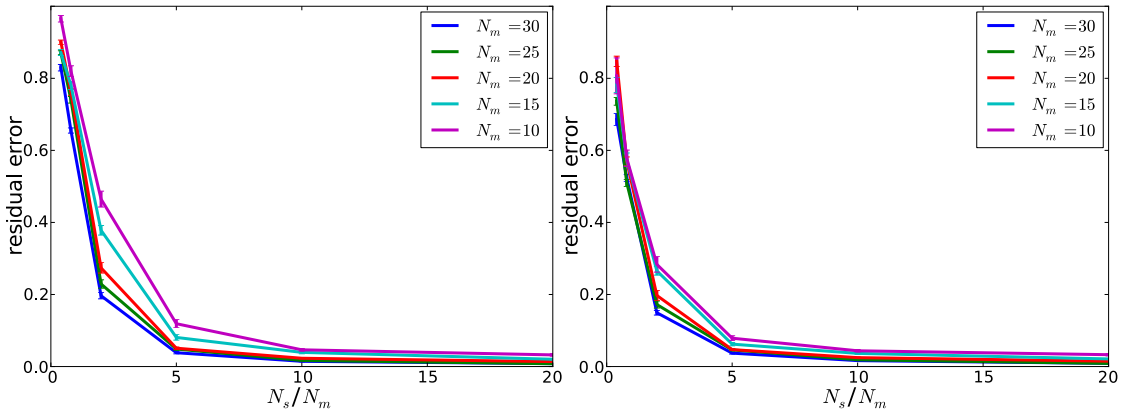


Figure 8.3: Residual error after learning in dependency on size ratio between sensory and motor populations for a complex world model with $N_w = 40$. Left: motor population, Right: sensory population.

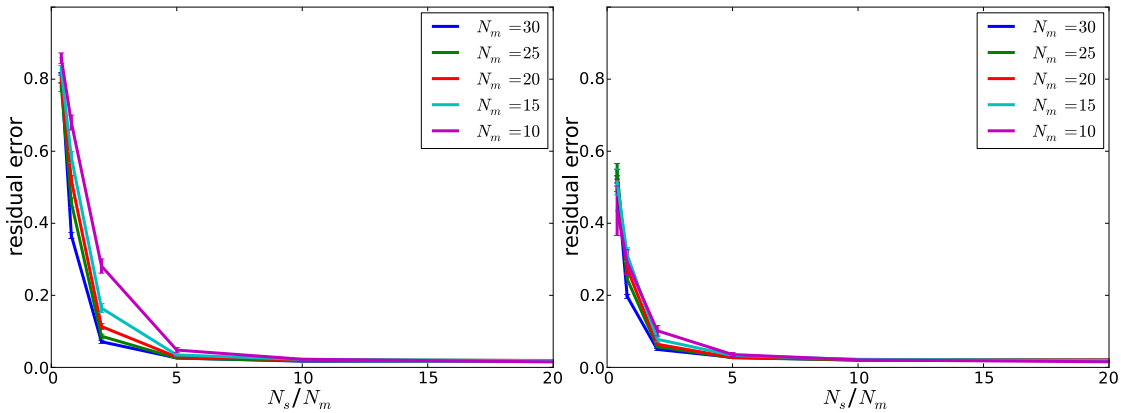
8.3.3 Dependency on System Size for Different World Models

This model was conceived to remedy the shortcomings of the simpler model introduced in chapter 5. Therefore, here the dependency on system size is tested for various world models with different temporal width.

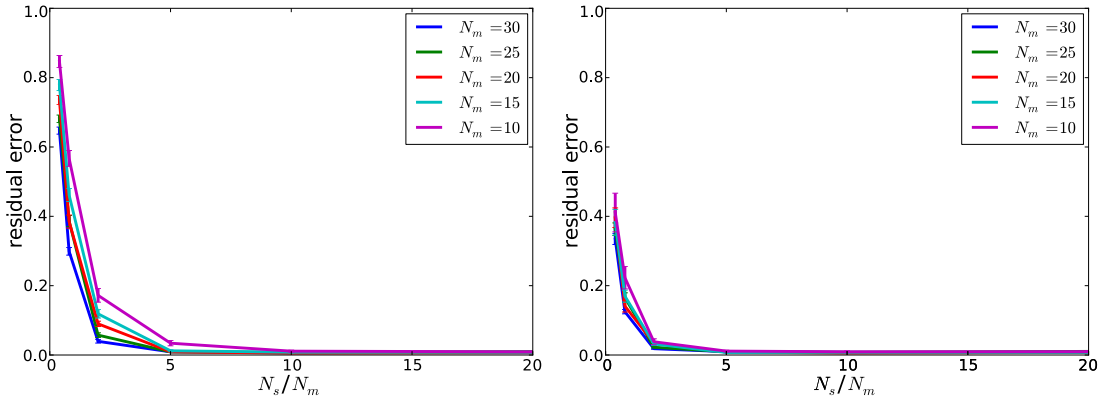
The dependency on system size is tested for $N_w = \{1, 5, 10, 40\}$. To explore how the residual error after learning depends on the system size, several different sizes are tested with $N_m = \{10, 15, 20, 25, 30\}$ for an exploration and song firing rate $r_{explor} = r_{test} = 1Hz$. Since it is reasonable to assume that the residual error will scale with the ratio $\alpha = N_s/N_m$ of neuron numbers in s and m, residual error after learning is computed for $\alpha = \{0.4, 0.8, 2, 5, 10, 20\}$. The results for different system sizes for different complexities of the world model are displayed in figures 8.3, 8.4(a), 8.4(b) and 8.4(c). For the most complex world model ($N_w = 40$), for low values of α in some cases cyclic activities between m and s_2m_2 occur, yielding very distances between tutor and recall pattern. Therefore, for the sake of visibility, the range of α is limited to $\alpha = \{2, 5, 10, 20\}$ for $N_w = 40$.
Learning is increasingly successful for increasing α. For simple world models, the learning success depends only very little on system size with lower residual error for larger systems (at low α). For the most simple world model, results are comparable to those from chapter 5, but here the residual error after learning increases less for increasing complexity of the world model. For the most complex model of the world with $N_w = 40$, however, there is a noticeable difference in the learning success, which much lower residual errors for larger systems. The slower decrease of the residual error for higher α for more complex world models is to be expected, because for more complex world models fewer neurons in s spike in the time interval which coincides with the teacher spike in s_2m_2. Therefore, fewer neurons contribute to learning at this point in time, hence increasing the necessary size difference between motor and sensory population to enable successful learning.



(a) Moderately complex world model with $N_w = 10$



(b) Moderately complex world model with $N_w = 5$



(c) Simple world model with $N_w = 1$

Figure 8.4: Residual error after learning in dependency on size ratio between sensory and motor populations for world models of different complexity. Left: motor population, Right: sensory population.

8.3.4 Dependency on Firing Rates

In the songbird, in most areas in the song system firing rates are higher than $1Hz$, so it is important to investigate learning also for higher firing rates. Due to the long after-hyperpolarisation in the motor area, spiking is prevented immediately after each spike for at least the time of the loop delay $\tau_{loop} = \tau_{ms} = 40ms$. Since the length of the hyperpolarisation in the motor population is chosen at $\tau_m = 70ms$ to provide a reasonably strong distance of the membrane potential to the threshold at the time of the self-generated input, the maximum firing rate is limited to at most $15Hz$, if the spiking is entirely regular. Here, the investigation of learning success is limited to the range of firing rates of $r_{explor} = r_{song} = \{1Hz, 2Hz, 3Hz, 4Hz, 5Hz, 6Hz, 7Hz\}$. To investigate how the success of learning depends on the motor firing rate, the residual error is calculated for different firing rates for $N_m = 20$ and $\alpha = \{10, 20\}$ and the complex world model with $N_w = 40$. The results are displayed in figure 8.5. Learning is successful for high α for low firing rates. For higher firing rates, the residual error rises to moderate levels.

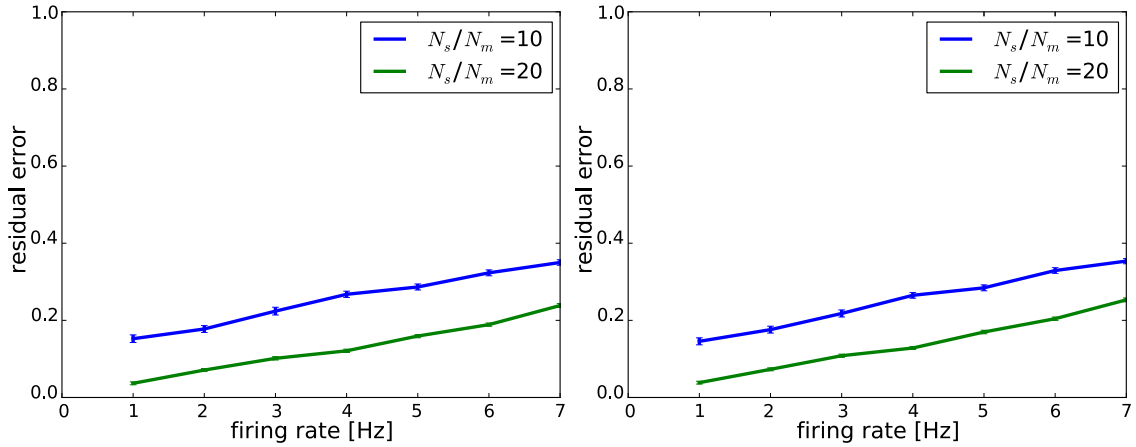


Figure 8.5: Residual error for different size ratios between sensory and motor populations in dependency of exploration firing rate. Song firing rate is equal to exploration firing rate. Left: motor population, Right: sensory population. The residual error is low for low firing rates and then rises to a moderate level with higher firing rates.

8.3.5 Necessity of Exploration with Testing Firing Rate

Up to this point, it was assumed that the firing rate in the motor population m was the same for both, the exploration phase and the song. Since this would require prior knowledge of the firing rate of the song, it is interesting to investigate, how the residual error changes when exploration is done with a firing rate different from the song firing rate. To this end, the ratio $\beta = r_{song}/r_{expl}$ is introduced. The residual error is measured for $N_m = 10$ and $N_s = 400$ for exploration firing rate $r_{expl} = \{1Hz, 2Hz, 3Hz\}$ and $\beta = \{0.25, 0.5, 1, 2, 4\}$. The results are displayed in figure 8.6.
Learning is successful, if the exploration rate is higher than the song firing rate, which is consistent with high firing rates during the exploration phase. Residual errors rise to moderate levels for high song firing rates and high exploration firing rates.

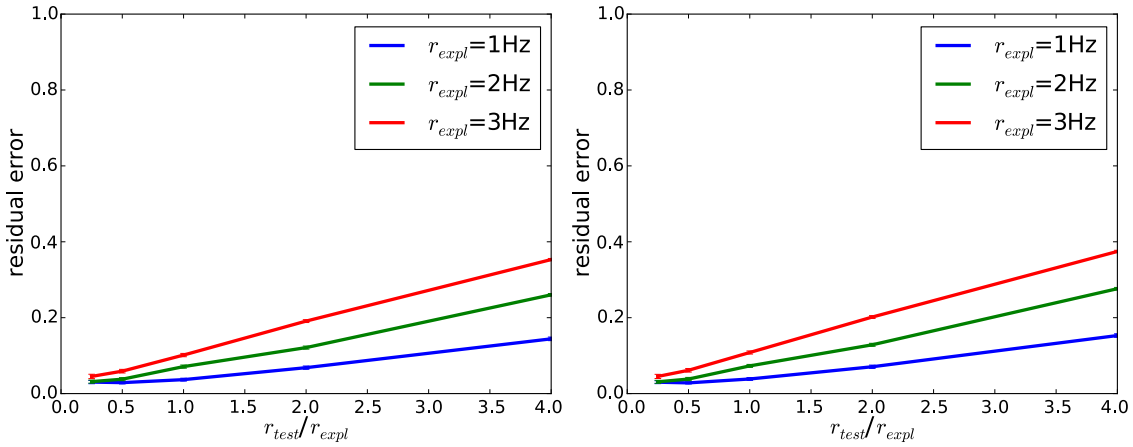
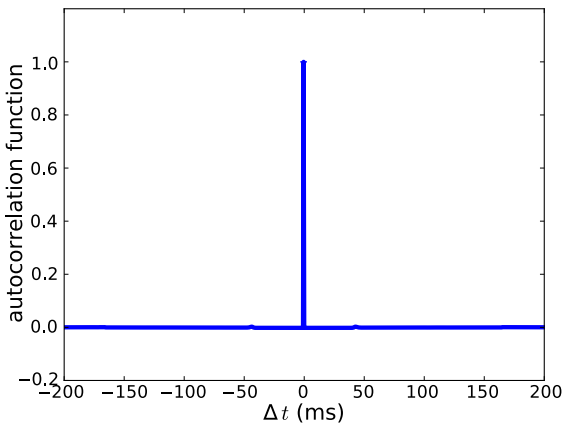


Figure 8.6: Residual error for different exploration firing rates in dependency on the firing rate ratios between song firing rate and exploration firing rate. Left: motor population, Right: sensory population. The residual error is low for low song firing rates for all exploration firing rate ratios and then rises to moderate level for higher ratios.

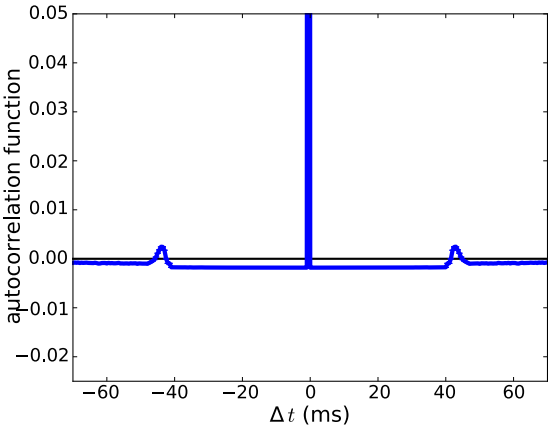
8.3.6 Experimentally testable Predictions

8.3.6.1 Autocorrelation Function

As for the simple inverse model presented in chapter 5, the long after-hyperpolarisation in the motor population in the model serves to avoid cyclic activity. A side effect of this hyperpolarisation is that the maximum firing rate in the motor population is limited, because immediately after a spike, each neuron is reset to a low membrane potential and thus has a very low spiking probability. This can be measured by measuring the autocorrelation of the spike trains. Figure 8.7 shows the autocorrelation of the spike trains in m for $N_m = 10$, $N_s = 200$, $r_{expl} = r_{song} = 7Hz$. There is a small, but noticeable dip in the autocorrelation of the length of the time of the loop delay. The autocorrelation could be accessible experimentally.



(a) Autocorrelation function of spiking activity in the motor population.



(b) Close-up of the autocorrelation function around $\Delta t = 0$ and $\rho(\Delta t) = 0$.

Figure 8.7: Autocorrelation function of the spiking activity in m. There is a dip of a width equivalent to the loop delay. The horizontal black line at autocorrelation 0 serves as a guide to the eye.

8.3.6.2 Mirror Neurons without Delay between Song and Playback

All of the above assumes that the neurons in s_2m_2 all receive their input from s_{recall}. The bird's brain, however, may also contain neurons in adjacent area s_xm_x that receive their inputs from the primary sensory area s_{sens}, as well as from the motor area m. Let us assume that here the same setting applies, i.e. that the feedback from m is delayed and the connections from s_{sens} to s_xm_x are plastic according to the same learning mechanism. Then, after learning, the neurons in s_xm_x indeed respond at similar times during both, active singing and passive listening, i.e. they represent zero delay mirror neurons as found experimentally in area HVC$_x$ ([PPNM08]) (see figure 8.3.6.2).

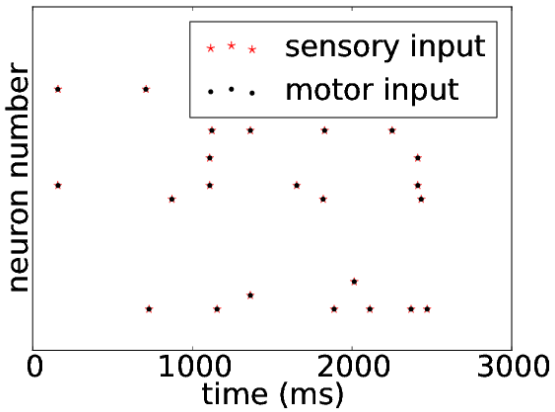


Figure 8.8: Mirror neurons in s_xm_x are active at approximately the same time relative to the song during singing and during passive playback.

8.4 Discussion

In this chapter, I show that the learning algorithm suggested for Chronotrons in chapter 7 can realize inverse models based on spatio-temporal spike patterns, even for complex models of the world. This model was conceived to remedy the limits of the model presented in chapter 5, which it does. In contrast to other learning rules operating on precise single spikes, this learning rule is biologically plausible, with a clear interpretation of the teacher signal. For a full discussion of the biological plausibility of the learning rule and its relationship to other learning rules, see 7.4.1 and 7.4.3.

The learning rule is applied in a new architecture consisting of coupled neuronal networks mimicking the anatomy of the songbird's brain to explain the reproduction of previously encountered acoustic signals. Crucially, this model assumes the existence of delayed feedback from a motor area to a sensorimotor region which serves as an intermediate relay for realizing the inverse model that has been suggested to underlie the bird's sound imitation capabilities. The existence of zero delay mirror neurons in birds [PPNM08] provides strong evidence for such a feedback.

The delay lines used in this model are a simplification of the possible delayed feedback in the songbird: In the songbird, these connections would not be monosynaptic. However, it can be assumed that this type of mixed delayed feedback is similar to the world model of temporal width of up to 10ms used in chapter 5 and can therefore be inverted with a mechanism of that type. Since the Anti-Hebbian STDP used in chapter 5 is compatible with the learning algorithm used here, I suspect that it could also serve to invert the inner delayed feedback loop from m to $s_2 m_2$ and back.

In this model, zero delay mirror neurons naturally emerge from the same learning mechanism acting on synapses that feed only sensory input, but not memory input, into areas that also receive delayed feedback. Interestingly, the singing related activity of the zero-delay mirror neurons found in HVC is not distorted by acoustic manipulations disrupting auditory feedback during singing, suggesting that these neurons receive purely motor-related input when the bird sings ([PPNM08]).

Furthermore, the model predicts mirror neurons with a delay equivalent to the loop delay in the respective sensory and motor areas involved in recall. This delay was linked to causal inverse models before ([HGH14]). In fact, experimental evidence for this type of delay was found, albeit in songbird brain area LMAN ([GKGH14]). In their study, [HGH14] suggest a simple Hebbian learning rule, which relies on the comparison between self-generated and target input. However, there is no clear biological interpretation of how this comparison could be achieved. Additionally, in their study, to allow for analytical tractability, they only discuss a linear model of sound generation and perception which is local in time. In the songbird, however, the process of sound generation and perception can be assumed to be both, non-linear, and non-local in time. In this thesis, non-linear spiking neurons are included for sound perception as well as for the motor-sensory mapping that includes interactions that are not local in time.

The model presented in this chapter requires a switch to suppress singing activity during passive playback, which could be implemented by strong inhibitory input into the motor area m. This hypothesis is supported by experimental evidence, since while in awake birds, motor neurons downstream of HVC are not responsive to auditory stimuli, in anaesthetized birds, playback of the bird's own song excites neurons in all nuclei in the song system downstream of HVC ([DM00, DK91, SWM03]). In the songbird, inhibitory

inputs into HVC are learned to suppress spiking activity during tutor song presentation [VKLL16]. Furthermore, the acoustically induced activation of mirror neurons is measured in sleeping or anaesthetized birds [PPNM08, GKGH14]. This implies that the visibility of mirror neurons in the songbird is an effect of sleep or anaesthesia.

In contrast to other models ([HGH14]), however, auditory input is not required to be gated off during singing, because echoing is suppressed by self-inhibition induced by long hyperpolarisations in the motor area m. This self-inhibition comes with a characteristic dip of the length of the loop delay in the spike autocorrelation of the involved motor neurons, which should be experimentally accessible.

The overall learning performance in this model for the most complex world model with $N_w = 40$ is worse than the learning performance in the model introduced in chapter 5. However, in chapter 5, only the most simple world model with a single delay line was investigated. In that model, learning breaks down completely for more complex world models. In contrast, the model presented in this chapter provides high learning performance even in the most complex world model. The analysis of the dependency on the firing rate in the motor population is only done for the most complex world model, but it can be expected that residual errors would be overall smaller for more simple world models as the one used in chapter 5.

Hence, together with the model from chapter 5, a complete setup for the learning of inverse models of complex world models was presented.

Chapter 9

Learning in Recurrent Networks with Membrane Potential Dependent Plasticity

In chapters 5 and 8, models for the learning of inverse models were introduced as a method for vocal learning. These inverse models serve to map a given sequence of sensory activations onto the matching motor activations. Before that can happen, however, a sensory activation sequence representing the tutor song has to be memorized. This chapter introduces a model for the imprinting of precise spatio-temporal sequences onto a recurrent network of spiking neurons. To that end, the algorithm for learning precisely timed spikes in response to spatio-temporal input patterns introduced in chapter 7 is applied to all connections in a recurrently coupled all-to-all network of spiking neurons.

9.1 Introduction

The generation of precise sequences of activation in neural networks is at the basis of almost all behaviour in animals. For example, it was found that songbirds produce extremely sparse and precise activations in HVC during singing [HKF02]. At least some of these sequences are a result of learning. Learning to reproduce a given spatio-temporal sequence of activations is a fundamental challenge, which has been addressed in early modelling studies (e.g. [Hop82]). However, previous models incorporated neither spiking neurons, nor biologically plausible learning rules.

The setup for each single neuron in a recurrent network is similar to the setting in the Chronotron problem: The neuron needs to learn to generate precisely timed spikes in response to an input pattern from all other neurons. Therefore, it is reasonable to try to apply the same learning rules that work for Chronotrons to the learning in recurrent networks.

However, recurrent networks are very sensitive to noise, such that slight deviations from the target pattern at a given time have large impact on the spiking pattern at later times. This behaviour was successfully reined in with the addition of noise during the learning process in [LB13]. There, robust spiking sequences in a network of rate neurons

were produced. Learning was started from sequences the networks would produce from the weight distribution at initialization, and these sequences were stabilized by learning with noise. Noise during learning is beneficial for the stability of recall.

The learning of sequences in recurrent networks has been treated with learning algorithms that were devised to solve the Chronotron problem before in [MRÖS14]. There, the FP-learning algorithm (for a brief description see section 3.4.2.3.3) was shown to be able to imprint several sequences onto recurrent networks of spiking neurons. However, FP-learning suffers from a lack of biological plausibility.

Here, the biologically plausible learning algorithm introduced in chapter 7 is applied to learning in recurrent networks, where it enables neurons to learn precisely timed sequences.

9.2 The Model

9.2.1 Network setup

To investigate whether the same mechanism used for learning precisely timed spikes as a response to spatio-temporal input patterns in the Chronotron setting in chapter 7 and the inverse model in chapter 8 can also serve to imprint spatio-temporal patterns on recurrent networks, an all-to-all network of N neurons is considered (see figure 3.1(b)). During learning, all synapses within the network are considered to be plastic and to obey (9.6) (see below). For proof of principle, the desired spatio-temporal patterns are taken to consist of N equidistant spikes with a distance of $d = 2ms$, where the order of spikes is randomly assigned. Each pattern is looped twice during each learning epoch lasting $T = 2Nd$ to allow for cyclic recall. All P patterns are shown to the network in batch mode and then the synaptic weights are updated. The patterns to be memorized are fed into the network by giving the respective neurons an input which mimics a synaptic input from a different neuron population with weight $a = 0.3$:

$$I_{ext} = a \exp\left(-\frac{t - t_{post}}{\tau_s}\right) \Theta(t - t_{post}). \tag{9.1}$$

During learning, all neurons receive additional gaussian noise of standard deviation $\sigma = 0.1mV$. Recall is performed without noise.

9.2.2 Neuron Model

In this chapter, as in chapter 5 and 8, simple integrate-and-fire neurons are used. For the sake of completeness, the description is repeated here.

All neurons are modelled as a simple leaky integrate-and-fire neuron. To facilitate the derivation of the plasticity rule, the formulation of the SRM_0 [GK02] is used. A neuron j receives input from other neurons i via plastic synapses of weight w_{ji}. The neuronal voltage $V(t)$ is given by the sum of weighted synaptic input kernels $\varepsilon(s)$ (postsynaptic potentials, PSPs) and reset kernels $R(s)$, which model the neuronal reset after a spike. External input currents $I_{ext}(t)$ are low-pass filtered with a response kernel $\kappa(s)$. The full

equation reads

$$V_j(t) = \sum_i w_{ji} \sum_k \varepsilon(t - t_k^i - t_{delay}) + \sum_{t_j} R(t - t_j) + \int_0^\infty \kappa(t-s) I_{ext}(s) ds. \tag{9.2}$$

Here, w_{ji} is the weight from presynaptic neuron i to the postsynaptic neuron j and t_i^k is the time of the k-th spike of presynaptic neuron with index i. A delay of synaptic transmission t_{delay} is included. The synaptic input kernel, the reset kernel and the passive filtering kernel are given by

$$\begin{aligned} \varepsilon(s) &= \Theta(s) \frac{1}{\tau_m - \tau_s} \left(\exp(-s/\tau_m) - \exp(-s/\tau_s) \right) && (9.3) \\ R(s) &= \Theta(s)(V_{reset} - V_{thr}) \exp(-s/\tau_m) && (9.4) \\ \kappa &= \exp\left(-(t-s)/\tau_m\right). && (9.5) \end{aligned}$$

where $\tau_m = 8ms$ is the membrane time constant of a LIF neuron determining the decay of voltage perturbations, and $\tau_s = 2ms$ is the decay time constant of synaptic currents, which defines the rise time of the PSP kernel. If there is no input, the voltage relaxes back to $V_{eq} = 0$. Spiking in this model is deterministic: If $V(t') = V_{thr} = 20mV$, the neuron spikes and a reset kernel is added at time $t' = t_j$. The formulation of the kernel makes sure that the voltage is always reset to $V_{reset} = -60mV < V_{eq}$.

9.2.3 Learning Rule

Since the learning in recurrent networks can be mapped onto the Chronotron problem as discussed in chapter 7, the same learning rule is used here. The description is repeated for the sake of completeness.

The plasticity rule is derived from the demand of a balanced membrane potential: the neuron should neither be hyperpolarized nor too strongly depolarized. This is a sensible demand, because it holds the neuron at a sensitive working point and keeps metabolic costs down. To that end, two thresholds are introduced, $\vartheta_P < \vartheta_D < V_{thr}$, between which the membrane potential is bounded. The weight change is chosen such that, whenever $\vartheta_D = 10mV$ is surpassed, all weights that contribute to the rise of the membrane potential are depressed, weighted by their respective influence given by the PSP-kernel ε. Whenever the membrane potential drops below $\vartheta_P = V_L$, all synapses that contribute to that downward deflection are potentiated, such that for a repetition of the pattern the membrane potential is deflected to stay within bounds. Additionally, the weights are bounded to stay below a maximum weight w_{max}, symbolizing a maximal synaptic strength. Limiting weights is advantageous for stability. The weight change is then given by

$$\dot{w}_{ji} = \eta \left(w_{max} - |w_i| \right) \left(-\gamma \left[V(t) - \vartheta_D \right]_+ + \left[\vartheta_P - V(t) \right]_+^2 \right) \sum_k \varepsilon \left(t - t_i^k - t_{delay} \right) . \tag{9.6}$$

$\gamma = 900$ is a factor that scales inhibition to excitation. The learning rate is given by $\eta = 0.1$.

9.2.4 Evaluation of Learning Success

To test whether the network is able to reproduce the target sequence, the first part of the target pattern is fed into the network, as during the learning process, to initialize

the network. The duration of the initialization period is $T_{init} = 30ms$. After learning, the network should be able to reproduce the target pattern, albeit possibly shifted and/or stretched. To evaluate the long-term stability of the pattern, the network runs for a longer time $T_{long-term} = fT$ with $f = 3$.

For the quantitative evaluation of the learning process, the spike train distance between the target pattern and the reproduced pattern is measured (see below) and learning success is evaluated (see below) every $\Delta N = 100$ learning epochs over a total of $N_k = 4000$ learning epochs for each pattern. Results are averaged over $n = 100$ sets of initializations consisting of P patterns. All weights are initialized as zero.

9.2.4.1 Measure of Pattern Similarity

To quantify the similarity of the target pattern and the self-generated pattern, it is necessary to compare two sets of spike trains for each pattern. The activity $a_i^{target}(t)$ in each neuron i in the tutor song will have to be compared to the activity during recall $a_i^{self}(t)$ to give some distance measure $d_0(a_i^{target}(t), a_i^{self}(t))$. The total distance over the activity a^{target} resp. a^{self} of all neurons in the given population will then just be the sum over all neurons in the population. Finally, this distance should be minimized over a global shift and a global stretching factor:

$$d(a^{target}, a^{self}) = \min_{\Delta t, c} \sum_i d_0(a_i^{target}(t), c \cdot a_i^{self}(t - \Delta t)) \tag{9.7}$$

The spikes during the initialization period are not taken into account. This distance is evaluated every $\Delta N_k = 50$ learning cycles. The resulting learning curves are normalized to the number of spikes in the target pattern, such that the distance before learning is 1.

9.2.4.2 Spike Train Distance Measures

There are several possible spike train distance measures $d_0(s_1, s_2)$, e.g. the VanRossum-distance [vR01] and the Victor-Purpura-distance [VP96].

To calculate the VanRossum-distance between two spike trains s_1 and s_2, both spike trains are convoluted with an exponential kernel. Then the quadratic distance is computed between those convolutions. While this spike train distance measure is easy to implement, it has the computational disadvantage of the computing time being dependent on the total number of simulation time steps.

Calculating the Victor-Purpura-distance seems more complicated, but is generally faster for not too high firing rates: To evaluate a spike train distance between spike trains s_1 and s_2, a cost for the transformation from s_1 into s_2 is calculated. There is a cost of 1 for the deletion or introduction of a spike and a cost of $q\Delta t$ for a shift of the spike time of one spike by Δt, where q is a parameter that scales the cost of shifting a spike relative to the insertion and deletion of spikes. The sum of the costs to transform s_1 into s_2 is then the spike train distance $d(s_1, s_2)$.

9.2.4.3 Measure of Learning Success

To evaluate if learning was successful in an individual trial, the distance of the recalled pattern to the target pattern in that trial and the number of spikes in the long-term recall are taken into account. If the patterns were reproduced perfectly and the recall was

perfectly stable, the patterns in the long-term recall condition should consist of $2fNP$ spikes. If the distance between recall and target pattern as introduced above is below 0.1 in the short term condition and the number of spikes in the long-term recall condition $N_{spikes}^{longterm}$ is between $0.9{\cdot}2fNP < N_{spikes}^{longterm} < 1.1{\cdot}2fNP$, the trial is counted as successful. This quantity can be normalized to the number of patterns in the trial P, thus yielding a success rate.

For the quantitative analysis, the fraction of correctly recalled patterns is computed every ΔN_k learning cycles. In each trial, the fraction of correctly recalled patterns after learning is computed by taking the average over the last 10% of measurements.

9.3 Results

In this chapter, learning of precise spatio-temporal spike patterns in recurrent networks is investigated. To that end, the learning mechanism introduced in chapter 7 is applied to an all-to-all coupled network of N neurons. P patterns of equidistant spikes are presented to the network via a strong teacher input given at the time of the desired spike. After successful learning, the network should reproduce the entire sequence when initialized with the beginning of the sequence. Furthermore, recall should be cyclic and therefore persist for times longer than the training period T.

9.3.1 Intuitive Understanding of the Learning Process

To facilitate an intuitive understanding of the learning process, for a single network of $N = 100$ neurons, which is supposed to learn $P = 3$ patterns, the spiking patterns at different stages of learning are presented and discussed. Note that for this size of the network, learning $P = 3$ patterns does not work in all cases, but the value is here chosen for better visibility of the patterns.

The extra input into a neuron around each teacher spike from the other neurons in the network emerging during learning will make the membrane potential steeper around the spike and cause the spikes to shift forwards in time during learning. However, during learning, the hyperpolarization, the signal for LTP, is slowly filled up, while at the same time the slope of the membrane potential before the spike becomes steeper. In net effect these two influences cancel each other out, so that learning comes to a halt.

Since after learning, around each spike for each neuron extra input is generated by the network, the network will fill in the missing teacher activity after learning, if the sequence is initialized by the first few spikes during an initialization period of $T_{init} = 30ms$. However, the resulting sequence may be stretched or shifted with respect to the original input sequence.

To test for stability of recall, it is checked if and for how long the network is able to reproduce the sequence. The recalled pattern is compared to both, the initial response of the network to the input before learning and the response of the network after learning. Since spikes may shift over time, the sequence slightly changes. Figure 9.2 shows raster plots of the different patterns imprinted on the same neuronal network for $t < T$ for different stages of learning.

Figure 9.3 shows a raster plot of the activity in the network after $N_k = 4000$ learning cycles during and after initialization for different patterns as it evolves for longer times. Recall is tested over six cycles. After learning, recall is cyclic and stable.

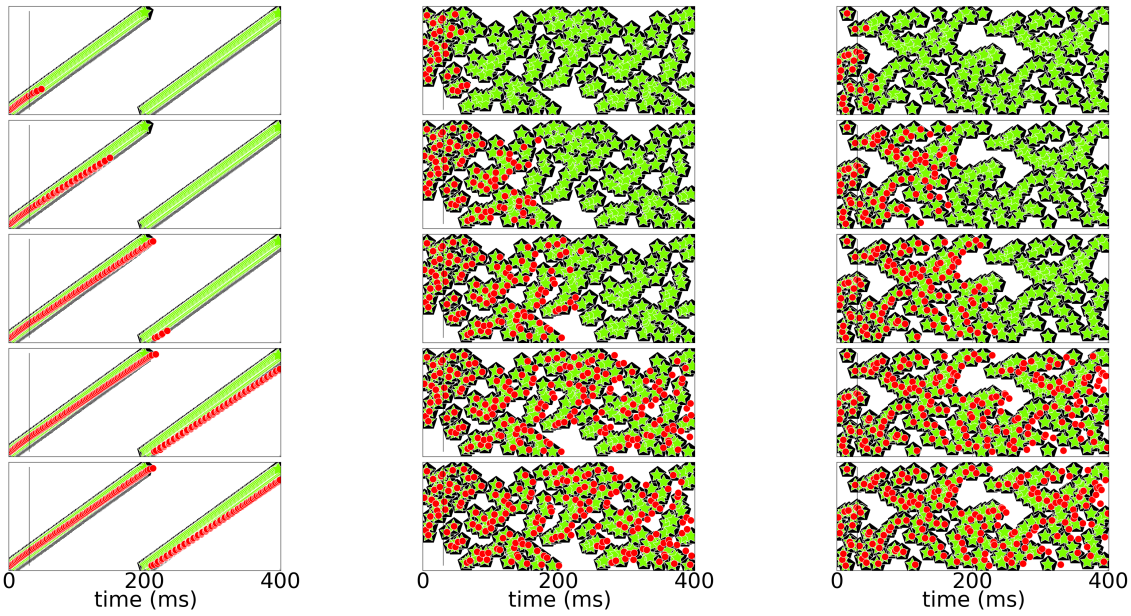


Figure 9.2: Spike raster plots at different stages of learning. Black pentagons are original activity of the network in response to the teacher input before learning, green stars are activity during learning and red dots are recall; the vertical black line demarks the end of the initialization sequence. From top to bottom after 50, 100, 150, 350 and 1200 learning cycles.

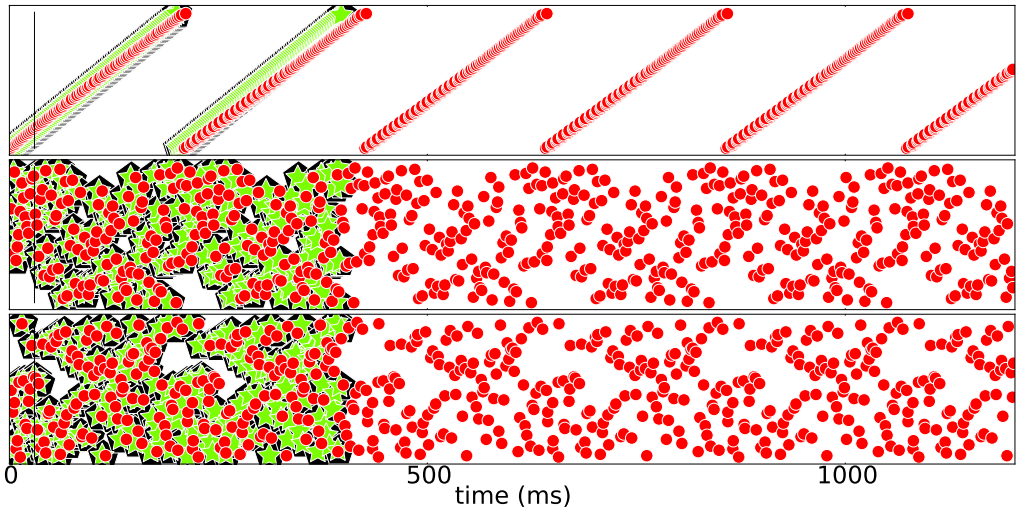
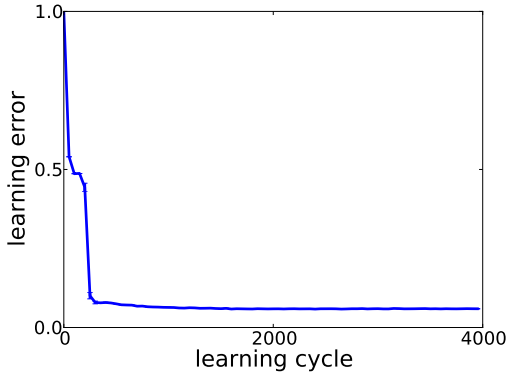


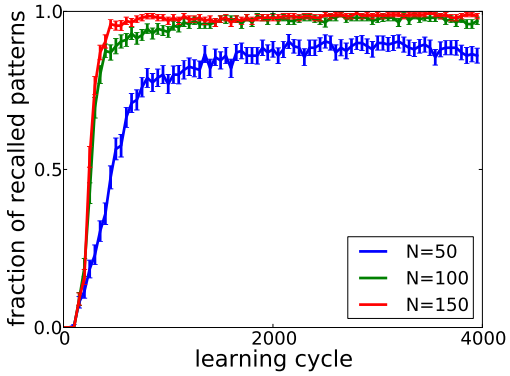
Figure 9.3: Spike raster plots for long times after learning. Black pentagons are original activity of the network in response to the teacher input before learning, green stars are activity during learning and red dots are recall; the vertical black line marks the end of the initialization sequence. The learned patterns are cyclic and stable after learning.

9.3.2 Quantitative Evaluation of the Learning Process

For the quantitative evaluation of the learning process, the spike train distance between target pattern and the reproduced pattern is measured every $\Delta N_k = 50$ learning epochs over a total of $N_k = 4000$ learning epochs for each pattern. The normalized distance between the recall and the target pattern (i.e. the original response of the network to the teacher input before learning) for $N = 50$ and $P = 1$, averaged over $n = 100$ sets of initializations, are displayed in figure 9.4(a).
Furthermore, a success measure for individual patterns is introduced: In individual trials a pattern is counted as successfully recalled, if the distance between recalled and target pattern is below 0.1 and the number of spikes in a recall pattern observed during $3T$ is within 10% of the target number of spikes during this time period. This quantity can then be normalized to the number of patterns P, yielding a fraction of correctly recalled patterns. The fraction of correctly recalled patterns over the course of learning for $N = \{50, 100, 150\}$ for $P = 2$ is displayed in figure 9.4(b). Learning converges quickly with success rates of close to 1 for sufficiently large networks.



(a) Distance between the self-generated pattern and the target pattern, optimized for a global stretch and shift averaged over patterns and initializations for $P = 1$ patterns in a network of $N = 50$ recurrently coupled neurons. The distance decays quickly in the learning process and settles on a low level.



(b) Fraction of correctly recalled patterns over the course of learning for $P = 2$ patterns in a network of $N = \{50, 100, 150\}$ recurrently coupled neurons. Learning is counted as successful, if in the individual trial the distance between the self-generated pattern and the target pattern is below 10% and the number of self-generated spikes for the long-term trial is within a 10% range around the expected number. Learning converges quickly with the fraction of correctly recalled patterns close to 1 for sufficiently large networks.

9.3.3 Scaling with the System Size

To investigate how the number of patterns a network can learn to reproduces scales with the system size, the success rate after learning is computed for individual trials as the average over the last 10% of measurements. From this quantity the average and standard error are computed.

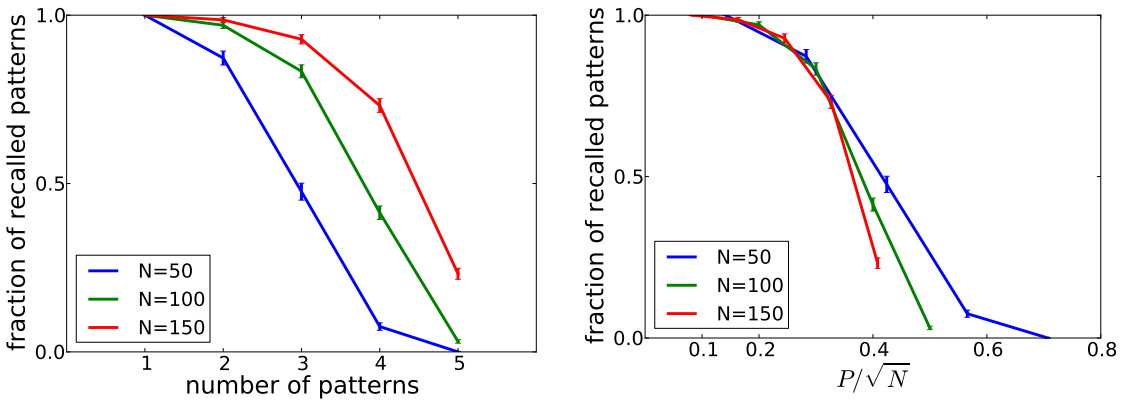


Figure 9.4: Left: Number of correctly recalled patterns in dependency of the number of patterns learned for different system sizes. Right: If the number of patterns P is normalized by the square root of the number of neurons in the network N, curves for different system sizes coincide. Learning is successful up to a critical load of about $\alpha^c_{\surd} = \frac{P}{\sqrt{N}} = 0.2$.

The ability to learn $P = \{1, 2, 3, 4, 5\}$ is evaluated for network sizes $N = \{50, 100, 150\}$ for $n = 100$ initializations. Learning works well if the number of patterns P is sufficiently small in comparison to the number of neurons N. When the fraction of correctly recalled patterns is plotted as a function of a load parameter $\alpha_{\surd} = \frac{P}{\sqrt{N}}$, the curves almost coincide for different system sizes. Learning is successful for $\alpha_{\surd} = \frac{P}{\sqrt{N}} < 0.2$ and breaks down above that. The slope of the curve around the cutoff point becomes steeper for larger networks, suggesting that for sufficiently large networks there may be a critical load, below which all patterns are learned and above which learning fails. However, since all networks considered here are relatively small, finite size effects may have an influence on the scaling behaviour.

9.4 Discussion

In this chapter, the plasticity mechanism introduced in chapter 7 was applied to the learning of elongated spatio-temporal sequences in recurrent networks. It was shown that it is possible to imprint several patterns onto a recurrent network, such that they will be reproduced in a stable manner, if the network is initialized with the beginning of the sequence.

The introduction of noise during learning has been shown to be beneficial for stability [LB13]. In their study, Laje and Buonomano showed that in a network of rate neurons, stable activation patterns can be learned, if innate activation patterns of the network were additionally trained with the addition of noise. This generates an attractor around the desired activation pattern sequence. Using innate patterns that the network would generate spontaneously before learning has the added benefit of guaranteeing that the pattern can in fact be learned. In this chapter, the slight variability of the spiking pattern due to the shifting of individual spikes during the learning process, which is a result of a real

teacher input, may be similarly beneficial for the selection of patterns that can be learned. In the model presented here, additive noise is used as well for the stabilization of the recall.

In this chapter, an algorithm that was originally devised for Chronotron learning was applied to the learning in recurrent networks. This is based on the insight that for each individual neuron, the situation in both setups is very similar: The neuron has to produce precisely timed spikes in response to a given input pattern, which in the case of recurrent networks is generated by the rest of the network.
Another algorithm devised for Chronotron learning (FP-learning) has been successfully applied to the learning in recurrent networks before [MRÖS14]. There, the finite precision that is required during the learning process allows the spikes to slightly move around during the learning process, which may be similarly effective for stabilizing sequences as the addition of noise is.

The interplay between the neuronal dynamics in the form of the hyperpolarization and the learning rule is particularly important for the learning mechanism. In their study, Brea et al. [BSP13] similarly propose that the learning algorithm should match the neuronal dynamics. They devise a learning rule starting from the given dynamics of the neuron, which optimizes the recall properties after learning. However, their learning algorithm only serves to approximate a given target spiking distribution, and thus does not operate on single spikes.

Both, the learning rule proposed by Brea et al. [BSP13] and FP-Learning [MRÖS14] suffer from an unclear interpretation of the teacher input. Here, in contrast, the teacher input is just a regular synaptic input, albeit of high amplitude. For a full discussion of the biological plausibility of MPDP in contrast to other learning rules, including the ones mentioned above, see 7.4.1 and 7.4.3.

In this chapter, only the arguably simplest type of pattern is investigated. However, since it is possible to imprint several such patterns at the same time, it is likely that more complex patterns can be learned as well. The number of patterns seems to grow with the square root of the number of neurons in the network. This is plausible: learning of individual spikes becomes simpler for growing networks sizes, because the pattern becomes sparser at any given point in time. However, learning becomes more difficult for growing network sizes, because the length of each pattern grows with the network size. The net effect is a scaling of the number of patterns that can be stored with the square root of the network size. However, in this work, only small networks are considered, such that the results may be influenced by finite size effects. It would be tempting to investigate the scaling behaviour more deeply and possibly even gain an analytical understanding of the scaling behaviour.

Chapter 10

Discussion

This thesis aims to shed light on the neuronal mechanisms that enable communication. In order to verbally communicate complex content, sound has to be associated with meaning. Furthermore, the respective sounds have to be reproduced to actively communicate. To that end, a sensory memory of the activation pattern in an auditory brain region needs to be formed. Finally, this sensory activation pattern needs to be translated into motor commands that reproduce the original sound.

In this chapter, I will provide a summary of the results in this thesis and how they contribute to answer the questions raised above and briefly discuss the results. For a more detailed discussion of each part, see the individual chapters. Finally, I will provide an outlook onto possible future work based on this thesis.

10.1 Summary and Discussion

In this thesis, two models for associative learning are introduced in chapter 6 and chapter 7. In the brain, associative learning may serve to associate two stimuli or to associate meaning with a given neuron activation pattern, e.g. the auditory representation of a sound. Thus, this type of associative learning is required for communication of complex content.

In both chapters, associative learning is discussed in a single layer feed-forward network. In the former, the associative learning of either spiking or not spiking in response to a purely spatial input pattern is discussed. It is shown that reverse spike-timing dependent plasticity (RSTDP) in combination with a hyperpolarization after each spike is mathematically equivalent to the Perceptron learning rule (PLR). This implies that the resulting learning algorithm has the beneficial properties of the PLR: a stop condition to avoid overlearning and a margin, which guarantees correct output also in the presence of some noise. The nature of the learning algorithm is pairing a hyperpolarization induced by a teacher spike with a synaptic learning rule, that raises the membrane potential of the output neuron, until a plasticity threshold below the spiking threshold is reached. The extra input that is generated to fill the hyperpolarization leads to a spike in the absence of the teacher spike.
In chapter 7, the Chronotron problem is discussed. Here, the output neuron in a sim-

ple feed-forward network is required to spike at a precisely defined time in response to a spatio-temporal input spike pattern. The learning mechanism follows the same principle as above: A teacher spike is induced at the desired spike time, which leads to a strong hyperpolarization. This hyperpolarization serves as a learning signal for the synaptic learning rule, which has the objective to balance all inputs into the output neuron and thus keep the membrane potential bounded. Thus, the hyperpolarization leads to a strong learning signal, which changes the weights to bend the membrane potential upwards for a repetition of the pattern. This process stops when enough extra input from the input pattern is generated for the membrane potential to hit the upper plasticity threshold (which is below the spiking threshold). At the same time, the membrane potential before the spike rises above that plasticity threshold, which induces a synaptic change bending the membrane potential towards more negative values for a repetition of the pattern. After the hyperpolarization is sufficiently filled up, these effects cancel each other out, such that plasticity comes to a halt. The extra input generated by the input pattern again leads to an output spike in the absence of the teacher spike, which is close in time to the original teacher spike time. Thus, the Chronotron problem is solved. Furthermore, the algorithm provides a mechanism to distinguish desired teacher spikes from undesired spurious spikes: weak input, which leads to a shallow rise before a spurious spike, will be suppressed, because the membrane potential stays above the upper plasticity threshold before the spike for a long time, giving a strong learning signal to suppress the spike. For stronger teacher inputs, the membrane potential rises steeply before the spike, thus limiting the time it stays above the plasticity threshold, which leads to a net strengthening effect. Furthermore, the time course of the membrane potential around the spike changes over the course of learning: It becomes steeper due to the extra input after the spike and the negative input before the spike, such that the resulting output spikes become more resistant to noise in the input pattern.

Compared to existing supervised learning rules, both of the learning rules introduced here are unique in that they provide a local synaptic update rule and a clear interpretation of how a teaching signal could be communicated to an individual neuron in the brain. The key parts of both learning rules are all biologically plausible, albeit not yet experimentally found. However, the sensitivity of the synaptic weight change in response to presynaptic spiking activity and postsynaptic sub-threshold depolarization found in experimental studies [ABS90, STN04, FDV09] points to biological mechanisms to transfer all relevant signals to the synapse, such that the update rule suggested in chapter 7 could be in place in real neurons.

In chapter 9, the latter mechanism is applied to the learning of sequences of spike patterns in recurrent networks. This process may be involved in the formation of the sensory memory of a given sound sequence. The application of a Chronotron learning algorithm to learning in recurrent networks is based on the insight that for each individual neuron involved in sequence learning in a recurrent network the learning task is the same as in the Chronotron problem: The neuron is required to produce a precisely timed spike in response to a given spatio-temporal input pattern, which is provided by the other neurons in the case of the recurrent network. It is shown that it is possible to imprint several spatio-temporal spiking patterns onto recurrent networks. The plasticity mechanism includes real teacher inputs, which allow the target spikes to shift slightly over the course

of learning. This allows the network to slightly modify the sequence towards sequences which are easier to reproduce. Furthermore, over the course of learning, the entire spike pattern is compressed in time, each time shifting to slightly earlier times relative the the preceding spikes, which allows spikes to be reproduced in the absence of the teacher spike. The addition of noise during the learning process is beneficial for stability of the learned patterns. After learning, recall is cyclic and stable for long times. In the context of this thesis, only learning of the simplest possible spiking patterns were investigated. However, the number of patterns, in which each neuron spikes exactly once, grows with the square root of the number of neurons in the network. Due to the fact that several such patterns can in principle be learned at the same time, it can be assumed that longer, more complex patterns can also be learned. Furthermore, it is likely that the plasticity mechanism presented here may enable recurrent neuronal networks to learn to reproduce less regular sequences as they would occur for sensory memory formation. A general difficulty for learning arbitrary sequences is posed by patterns that cannot be produced by recurrent networks, where all neurons are involved in the pattern in question: linearly non-separable patterns or patterns with too long time stretches of silence, in which the activity dies out. Adding hidden neurons, which are at liberty to fire at any given point in time unconstrained by a teacher, as used in [BSP13], might help to alleviate these constraints.

Finally, in chapters 5 and 8, vocal learning in the context of inverse models is discussed. The basic concept of inverse models is simple: During an exploration phase, the learner learns to invert the action perception cycle. Essentially, it generates random motor activations, which cause some sound and the thus perceived mapping “after this motor activation, this particular sound happens“ can be inverted to “if this is the target sound, this is the particular motor pattern required”. After this inverse model is learned, any sound sequence in the range of producible sounds can be produced by just feeding an auditory memory sequence into the inverse model, thus mapping the auditory sequence onto the motor sequence which in turn generates the same sound sequence.

In chapter 5, a simple model for learning inverse models is introduced: A system of connected neuron populations mimicking the brain architecture of songbirds is coupled with the learning mechanism also used for Perceptron learning in chapter 6. During the exploration phase, a motor population generates random activation patterns, which are translated into input into the sensory population via a model of the sound generation and perception process. After successful learning, the synaptic weights from the sensory to the motor population are adapted such that the extra input filling up the hyperpolarisation after each exploratory spike in the motor neurons drives the motor neurons to spike in the recall case. If learning is successful, the sensory patterns are thus mapped back onto the motor patterns that caused them. The sound generation and perception process is modelled as one or several weighted delay lines from the motor population to the sensory population. The learning of inverse models works well for simple models of the sound generation and perception process with a single fixed delay. Learning works well for a size difference between the sensory and the motor population, where the sensory population is at least five times bigger than the motor population. Due to the long hyperpolarization in the motor population, which serves to suppress cyclic recall and as a testbed for the input generated by the sensory population, the spike rate in the motor population is limited. The firing rate during exploration does not have to match the firing rate of the

tutor song, with learning being successful, if the exploration firing rate is higher than the song firing rate. Another interesting question is how robust learning in this setup is to unstructured background noise: Residual errors after learning remain small for a noise level in the sensory population up to 200% of the bird's own song. Another interesting consequence of the long hyperpoplarization is a dip in the spike autocorrelation function of the motor neurons of duration matching the loop delay from motor activation to sensory activation and back. This might be accessible experimentally. Finally, this type of inverse model entails mirror neurons in the motor and the sensory population, which are active in the same way during singing and during passive listening to the same song. The time shift between the two is of the same length as the loop delay. For more complex models of the sound generation and perception process, learning works still moderately well for a temporal spread of the input into the sensory population of up to 10ms. For more complex world models, however, for a width of the world model of 40ms, learning breaks down. Here, spatio-temporal spike patterns need to be mapped back onto precisely timed spikes, which the learning mechanism is unable to achieve.

Since this mapping back from spatio-temporal spike patterns onto precisely timed spikes is very similar to the Chrontoron problem, the learning algorithm suggested in chapter 7 is applied to learning of inverse models in chapter 8. The learning rule is applied in a new architecture consisting of coupled neuronal networks mimicking the anatomy of the songbird's brain. Crucially, this model assumes the existence of delayed feedback from a motor area to a sensorimotor region which serves as an intermediate relay for realizing the inverse model.

In this model, zero delay mirror neurons naturally emerge from the same learning mechanism acting on synapses that feed only sensory input but not memory input, into the sensorimotor areas that also receive delayed feedback. Furthermore, the model predicts mirror neurons with a delay equivalent to the loop delay in the respective sensory and motor areas involved in recall. The model requires a switch to suppress singing activity during passive playback, which could be implemented by strong inhibitory input into the motor area m. In contrast to other models [HGH14], however, no gating off of auditory input during vocalization is required.
The delay lines used in this model are a simplification of the possible delayed feedback in the songbird: In the songbird, these connections would not be monosynaptic. However, it can be assumed that this type of mixed delayed feedback is similar to the world model of temporal width of up to 10ms used in chapter 5 and can therefore be inverted with a mechanism of that type. Since the Anti-Hebbian STDP used in chapter 5 is compatible with the learning algorithm used here, I suspect that it could also serve to invert the inner delayed feedback loop from the motor area to the sensorimotor area and back.

Learning is successful for both, simple and complex models of the sound generation and perception process, albeit residual errors after learning are slightly higher for complex world models. For the complex model of the world, learning for different firing rates was tested and found to be successful for moderate firing rates. Firing rates during the exploration phase do not need to match firing rates of the target pattern. As in the simple model for inverse model learning introduced in chapter 5, the long hyperpolarisation in the motor population entails a limit on the maximal firing rate and a dip in the spike autocorrelation in these neurons. Again, this could be accessible experimentally, as well as the existence of mirror neurons.

In conclusion, in this thesis, models for associative learning which may serve to associate meaning with sound were introduced as well as an conceptually complete model for vocal learning composed of a novel architecture and a learning mechanism that not only serves to form inverse models, but is also able to imprint spiking sequences onto recurrent networks, which may underlie the formation of sensory memories.

10.2 Outlook and Future Work

Inverse model learning is an interesting approach to vocal learning. The inversion of the sound generation and perception process leads to an ability of the learner to imitate arbitrary sounds within the range of sounds it can produce at all. In this thesis, the investigation of inverse model learning was limited to relatively simple models of the sound generation and perception process, which is a simplification of the real processes involved. The sound generation process in zebra finches can be modelled as a surprisingly low-dimensional process [APMM13], making it accessible for theoretical studies. It would thus be interesting to develop a model for the sound perception process and the coding of motor gestures in the motor model, thus modelling the sound generation and perception process in more detail. It would then be interesting to see, how the results found in chapter 5 and 8 generalize to such a more complex model of this process.

From a theoretical and technical point of view, it would be interesting to test the abilities of inverse model learning as suggested in this thesis in setups involving physical sound generation modules. It should be possible to use any physical structure that produces sufficiently diverse sounds and a microphone, as well as a model for conversion of sound waves into input into the sensory population and a model for the conversion of activation in the motor population into physical movement in the sound generation module. In principle, the learning mechanism here should be able to invert any such mapping to some degree and thus produce arbitrary sounds (in the realm of sounds it can produce at all).

Finally, it would be interesting to investigate the properties of the recurrent network learning mechanism suggested in chapter 9 in more detail. The scaling behaviour with the square root of the network size is a peculiar effect, that could be investigated further numerically or even analytically. Furthermore, it would be interesting to investigate how recall is affected by noise. Moreover, the size and shape of the attractors around the target sequence could be investigated, as well as how these are influenced by this amplitude of noise during learning.

Last, but not least, the models devised for inverse model learning make the prediction of a dip in the autocorrelation function of the motor population involved. It could be investigated experimentally, if any of the (pre-)motor nuclei of the songbird have spiking statistics matching the requirements of the model. It could thus be tested experimentally, if the type of inverse model suggested here could indeed be in place in the songbird.

Numerical Procedures

Throughout this thesis, all networks were numerically integrated using a simple Euler integration scheme. The simulations were written in Python and used a step size of $0.5ms$ for all leaky integrate-and-fire neurons, a step size of $0.01ms$ for the conductance-based integrate-and-fire neurons and a step size of $0.005ms$ for the Hodgkin-Huxley type neurons. The values of neurons parameters can be found in the respective chapters.

Publications

Sascha Helduser, Maren Westkott, Klaus Pawelzik, and Onur Güntürkün. *The putative pigeon homologue to song bird lman does not modulate behavioral variability.* Behavioural brain research, 263:144148, 2014.

Christian Albers, Maren Westkott, and Klaus Pawelzik. *Perfect associative learning with spike-timing-dependent plasticity.* In C. J. C.Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, Advances in Neural Information Processing Systems 26, pages 17091717. Curran Associates, Inc., 2013.

Christian Albers, Maren Westkott, and Klaus Pawelzik. *Learning of Precise Spike Times with Homeostatic Membrane Potential Dependent Synaptic Plasticity.* PLoS ONE 11(2): e0148948. doi:10.1371/journal.pone.0148948, 2016.

Maren Westkott and Klaus Pawelzik. *An Integrative Account of Acoustic Sequence Imitation Learning in the Songbird.* Frontiers in Computational Neuroscience 2015 *under review.*

Acknowledgements

I have a lot of people to thank without whom the completion of this work would have been impossible. First and foremost, I have to thank my supervisor Klaus Pawelzik for all the creative input and all the new ideas, as well as for the fruitful discussions. I would like to thank Agnes Janßen for the relentless administrative support, which makes life in the group so much easier, as well as for the enthusiasm, good mood and cookies she provides for the group. For great technical support, I would like to thank David Rothermund, who is always available for the stupidest of questions, which he answers with remarkable calmness. I would like to thank all members of the group for providing a friendly and stimulating work environment. In particular, I am grateful to Christian Albers for the work we did together, for long fruitful discussions and for advice. I would like to thank my proof readers Christian Albers, Nergis Tömen, Federica Capparelli, Daniel Harnack and last but not least Axel Grzymisch, who greatly contributed to making this work more readable, understandable and overall better.

Finally, I would like to thank friends and family for the ongoing support over the course of the years. In particular, I am grateful to my partner M.L. for the ongoing support, feeding and bribing throughout the writing process.

Bibliography

[ABS90] A Artola, S Bröcher, and W Singer. Different voltage-dependent thresholds for inducing long-term depression and long-term potentiation in slices of rat visual cortex. *Nature*, 347(6288):69–72, 1990.

[AK90] LF Abbott and Thomas B Kepler. Model neurons: From hodgkin-huxley to hopfield. In *Statistical mechanics of neural networks*, pages 5–18. Springer, 1990.

[AL01] D Attwell and S B Laughlin. An energy budget for signaling in the grey matter of the brain. *Journal of Cerebral Blood Flow & Metabolism*, 21(10):1133–1145, 2001.

[APMM13] Ana Amador, Yonatan Sanz Perl, Gabriel B Mindlin, and Daniel Margoliash. Elemental gesture dynamics are encoded by song premotor cortical neurons. *Nature*, 495(7439):59–64, 2013.

[AWP13] Christian Albers, Maren Westkott, and Klaus Pawelzik. Perfect associative learning with spike-timing-dependent plasticity. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26*, pages 1709–1717. Curran Associates, Inc., 2013.

[AWP16] Christian Albers, Maren Westkott, and Klaus Pawelzik. Learning of precise spike times with membrane potential dependent synaptic plasticity. *PLoS ONE*, (11):(2): e0148948, 2016.

[BD02] Michael S Brainard and Allison J Doupe. What songbirds teach us about learning. 2002.

[BD13] Michael S Brainard and Allison J Doupe. Translating birdsong: songbirds as a model for basic and applied medical research. *Annual review of neuroscience*, 36:489, 2013.

[BKLP02] Sander M Bohte, Joost N Kok, and Han La Poutre. Error-backpropagation in temporally encoded networks of spiking neurons. *Neurocomputing*, 48:17–37, 2002.

[BMA84] Sarah W Bottjer, Elizabeth A Miesner, and Arthur P Arnold. Forebrain lesions disrupt development but not maintenance of song in passerine birds. *Science*, 224(4651):901–903, 1984.

[BSP13] Johanni Brea, Walter Senn, and Jean-Pascal Pfister. Matching recall and storage in sequence learning with spiking neural networks. *The Journal of Neuroscience*, 33(23):9565–9575, 2013.

[CBVG10] Claudia Clopath, Lars Büsing, Eleni Vasilaki, and Wulfram Gerstner. Connectivity reflects coding: a model of voltage-based STDP with homeostasis. *Nature Neuroscience*, 13(3):344–352, 2010.

[CD08] Natalia Caporale and Yang Dan. Spike timing-dependent plasticity: A hebbian learning rule. *Annu Rev Neurosci*, 31:25–46, 2008.

[DA01] Peter Dayan and Larry F Abbott. *Theoretical Neuroscience*. The MIT Press, 2001.

[DK91] Allison J Doupe and Masakazu Konishi. Song-selective auditory circuits in the vocal control system of the zebra finch. *Proceedings of the National Academy of Sciences*, 88(24):11339–11343, 1991.

[DK99] Allison J Doupe and Patricia K Kuhl. Birdsong and human speech: Common themes and mechanisms. *Annu. Rev. Neurosci*, 22:567–631, 1999.

[DLH10] Prashanth D'Souza, Shih-Chii Liu, and Richard H R Hahnloser. Perceptron learning rule derived from spike-frequency adaptation and spike-time-dependent plasticity. *Proceedings of the National Academy of Sciences*, 107(10):4722–4727, 2010.

[DM00] Amish S Dave and Daniel Margoliash. Song replay during sleep and computational rules for sensorimotor vocal learning. *Science*, 290(5492):812–816, 2000.

[DO87] Sigurd Diederich and Manfred Opper. Learning of correlated patterns in spin-glass networks by local learning rules. *Physical review letters*, 58(9):949, 1987.

[DP04] Yang Dan and Mu-Ming Poo. Spike timing-dependent plasticity of neural circuits. *Neuron*, 44(1):23–30, 2004.

[DP06] Yang Dan and Mu-Ming Poo. Spike timing-dependent plasticity: from synapse to perception. *Physiological reviews*, 86(3):1033–1048, 2006.

[DPFF+92] Giuseppe Di Pellegrino, Luciano Fadiga, Leonardo Fogassi, Vittorio Gallese, and Giacomo Rizzolatti. Understanding motor events: a neurophysiological study. *Experimental brain research*, 91(1):176–180, 1992.

[FD02] Robert C Froemke and Yang Dan. Spike-timing-dependent synaptic modification induced by natural spike trains. *Nature*, 416(6879):433–438, 2002.

[FDV09] Elodie Fino, Jean Michel Deniau, and Laurent Venance. Brief subthreshold events can act as Hebbian signals for long-term plasticity. *PLoS One*, 4(8), 2009.

[Fel00] Daniel E Feldman. Timing-based ltp and ltd at vertical inputs to layer ii/iii pyramidal cells in rat barrel cortex. *Neuron*, 27(1):45–56, 2000.

[FFS07] Ila R Fiete, Michale S Fee, and H Sebastian Seung. Model of birdsong learning based on gradient estimation by dynamic perturbation of neural conductances. *Journal of neurophysiology*, 98(4):2038–2057, 2007.

[Flo12a] Rzvan V. Florian. The chronotron: A neuron that learns to fire temporally precise spike patterns. *PLoS One*, 7(8), 2012.

[Flo12b] Rzvan V. Florian. The chronotron: A neuron that learns to fire temporally precise spike patterns. *PLoS ONE*, (7), 2012.

[FPD05] Robert C Froemke, Mu-Ming Poo, and Yang Dan. Spike-timing-dependent synaptic plasticity depends on dendritic location. *Nature*, 434(7030):221–225, 2005.

[FS06] Ila R Fiete and H Sebastian Seung. Gradient learning in spiking neural networks by dynamic perturbation of conductances. *Physical review letters*, 97(4):048104, 2006.

[FS10] Michale S Fee and Constance Scharff. The songbird as a model for the generation and learning of complex sequential behaviors. *ILAR journal*, 51(4):362–377, 2010.

[GK02] Wulfram Gerstner and Werner M Kistler. *Spiking Neuron Models: Single Neurons, Populations, Plasticity*. Cambridge university press, 2002.

[GKGH14] Nicolas Giret, Joergen Kornfeld, Surya Ganguli, and Richard HR Hahnloser. Evidence for a causal inverse model in an avian cortico-basal ganglia circuit. *Proceedings of the National Academy of Sciences*, 111(16):6063–6068, 2014.

[GM08] Tim Gollisch and Markus Meister. Rapid neural coding in the retina with relative spike latencies. *Science*, 319(5866):1108–1111, 2008.

[GS06] Robert Gütig and Haim Sompolinsky. The tempotron: a neuron that learns spike timing–based decisions. *Nature neuroscience*, 9(3):420–428, 2006.

[Heb49] Donald Olding Hebb. *The organization of behavior: A neuropsychological approach*. John Wiley & Sons, 1949.

[HGH14] A Hanuschkin, S Ganguli, and RHR Hahnloser. A hebbian learning rule gives rise to mirror neurons and links them to control theoretic inverse models. *Closing the Loop Around Neural Systems*, page 274, 2014.

[HKF02] Richard H R Hahnloser, Alexay A Kozhevnikov, and Michael S Fee. An ultra-sparse code underlies the generation of neural sequences in a songbird. *Nature*, 419:65–70, 2002.

[HKP91] John Hertz, Anders Krogh, and Richard G Palmer. *Introduction to the Theory of Neural Computation*, volume 1. Santa Fe Institute Series (Addison-Wesley Longman, Boston), 1991.

[HNA06] Julie S Haas, Thomas Nowotny, and Henry DI Abarbanel. Spike-timing-dependent plasticity of inhibitory synapses in the entorhinal cortex. *Journal of Neurophysiology*, 96(6):3305–3313, 2006.

[Hop82] John J Hopfield. Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences*, 79(8):2554–2558, 1982.

[Hop07] John J Hopfield. Hopfield network. *Scholarpedia*, 2(5), 2007. revision 91362.

[HTY+14] Kosuke Hamaguchi, Katherine A Tschida, Inho Yoon, Bruce R Donald, and Richard Mooney. Auditory synapses to song premotor neurons are gated off during vocalization in zebra finches. *Elife*, 3:e01833, 2014.

[HWPG14] Sascha Helduser, Maren Westkott, Klaus Pawelzik, and Onur Güntürkün. The putative pigeon homologue to song bird lman does not modulate behavioral variability. *Behavioural brain research*, 263:144–148, 2014.

[HZK10] Knut Holthoff, Dejan Zecevic, and Arthur Konnerth. Rapid time course of action potentials in spines and remote dendrites of mouse visual cortex neurons. *The Journal of Physiology*, 588(7):1085–1096, 2010.

[ID03] Eugene M Izhikevich and Niraj S Desai. Relating stdp to bcm. *Neural computation*, 15(7):1511–1523, 2003.

[KB06] Mimi H Kao and Michael S Brainard. Lesions of an avian basal ganglia circuit prevent context-dependent changes to song variability. *Journal of neurophysiology*, 96(3):1441–1455, 2006.

[KDB05] Mimi H Kao, Allison J Doupe, and Michael S Brainard. Contributions of an avian basal ganglia–forebrain circuit to real-time modulation of song. 2005.

[LB13] Rodrigo Laje and Dean V Buonomano. Robust timing and motor patterns by taming chaos in recurrent neural networks. *Nature neuroscience*, 16(7):925–933, 2013.

[Leo04] Anthony Leonardo. Experimental test of the birdsong error-correction model. *Proceedings of the National Academy of Sciences of the United States of America*, 101(48):16935–16940, 2004.

[LHS+07] Karri P Lamsa, Joost H Heeroma, Peter Somogyi, Dmitri A Rusakov, and Dimitri M Kullmann. Anti-Hebbian long-term potentiation in the hippocampal feedback inhibitory circuit. *Science*, 315(5816):1262–1266, March 2007.

[LNM05] Robert Legenstein, Christian Naeger, and Wolfgang Maass. What can a neuron learn with spike-timing-dependent plasticity? *Neural Computation*, 17(11):2337–2382, 2005.

[Mar70] Peter Marler. Birdsong and speech development: Could there be parallels? there may be basic rules governing vocal learning to which many species conform, including man. *American scientist*, 58(6):669–673, 1970.

[Mas13] Timothee Masquelier. Neural variablility, or lack thereof. *Frontiers in computational neuroscience*, 7, 2013.

[MLFS97] Henry Markram, Joachim Lübke, Michael Frotscher, and Bert Sakmann. Regulation of synaptic efficacy by coincidence of postsynaptic APs and EPSPs. *Science*, 275(5297):213–215, 1997.

[Moo09] Richard Mooney. Neural mechanisms for learned birdsong. *Learning & Memory*, 16(11):655–669, 2009.

[Moo14] Richard Mooney. Auditoryvocal mirroring in songbirds. *Phil. Trans. R. Soc. B*, 369:20130179, 2014.

[MP69] Marvin Minsky and Seymour Papert. Perceptrons. 1969.

[MRÖS14] Raoul-Martin Memmesheimer, Ran Rubin, Bence P Ölveczky, and Haim Sompolinsky. Learning precisely timed spikes. *Neuron*, 82(4):925–938, 2014.

[MS95] Zachary F Mainen and Terrence J Sejnowski. Reliability of spike timing in neocortical neurons. *Science*, 268(5216):1503–1506, 1995.

[NSL76] Fernando Nottebohm, Tegner M Stokes, and Christiana M Leonard. Central control of song in the canary, serinus canarius. *Journal of Comparative Neurology*, 165(4):457–486, 1976.

[ÖAF05] Bence P Ölveczky, Aaron S Andalman, and Michale S Fee. Vocal experimentation in the juvenile songbird requires a basal ganglia circuit. *PLoS Biol*, 3(5):e153, 2005.

[PGCZ14] Marko A Popovic, Xin Gao, Nicholas T Carnevale, and Dejan Zecevic. Cortical dendritic spine heads are not electrically isolated by the spine neck from membrane potential signals in parent dendrites. *Cerebral Cortex*, 24(2):385–395, 2014.

[PK10] Filip Ponulak and Andrzej Kasinski. Supervised learning in spiking neural networks with ReSuMe: sequence learning, classification, and spike shifting. *Neural Computation*, 22(2):467–510, 2010.

[PPNM08] Jonathan F Prather, S Peters, S Nowicki, and R Mooney. Precise auditory–vocal mirroring in neurons for learned vocal communication. *Nature*, 451(7176):305–310, 2008.

[PS09] Lucy M Palmer and Greg J Stuart. Membrane potential changes in dendritic spines during action potentials and synaptic input. *The Journal of Neuroscience*, 29(21):6897–6903, 2009.

[Ros58] Frank Rosenblatt. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6):386, 1958.

[RPMJ04] Anton Reiner, David J Perkel, Claudio V Mello, and Erich D Jarvis. Songbirds and the revised avian brain nomenclature. *Annals of the New York Academy of Sciences*, 1016(1):77–108, 2004.

[SB06] Jon T Sakata and Michael S Brainard. Real-time contributions of auditory feedback to avian vocal motor control. *The Journal of neuroscience*, 26(38):9619–9628, 2006.

[SBC02] Harel Z Shouval, Mark F Bear, and Leon N Cooper. A unified model of NMDA receptor-dependent bidirectional synaptic plasticity. *Proceedings of the National Academy of Sciences*, 99(16):10831–10836, 2002.

[SH06] Per Jesper Sjöström and Michael Häusser. A cooperative switch determines the sign of synaptic plasticity in distal dendrites of neocortical pyramidal neurons. *Neuron*, 51(2):227–238, 2006.

[SMA00] Sen Song, Kenneth D Miller, and Larry F Abbott. Competitive hebbian learning through spike-timing-dependent synaptic plasticity. *Nature Neuroscience*, 3(9):919–926, 2000.

[SN91] Constance Scharff and Fernando Nottebohm. A comparative study of the behavioral deficits following lesions of various parts of the zebra finch song system: implications for vocal learning. *The Journal of neuroscience*, 11(9):2896–2913, 1991.

[STN01] Per Jesper Sjöström, Gina G Turrigiano, and Sacha B Nelson. Rate, timing, and cooperativity jointly determine cortical synaptic plasticity. *Neuron*, 32(6):1149–1164, 2001.

[STN04] Per Jesper Sjöström, Gina G Turrigiano, and Sacha B Nelson. Endocannabinoid-Dependent Neocortical Layer-5 LTD in the Absence of Postsynaptic Spiking. *Journal of Neurophysiology*, 92(6):3338–3343, August 2004.

[SWM03] Christopher B Sturdy, J Martin Wild, and Richard Mooney. Respiratory and telencephalic modulation of vocal motor neurons in the zebra finch. *The Journal of neuroscience*, 23(3):1072–1086, 2003.

[TBW+11] John A Thompson, Mark J Basista, Wei Wu, Richard Bertram, and Frank Johnson. Dual pre-motor contribution to songbird syllable variation. *The Journal of Neuroscience*, 31(1):322–330, 2011.

[US14] Robert Urbanczik and Walter Senn. Learning by the dendritic prediction of somatic spiking. *Neuron*, 81:521–528, 2014.

[VGO+13] Matthijs B Verhoog, Natalia A Goriounova, Joshua Obermayer, Jasper Stroeder, JJ Johannes Hjorth, Guilherme Testa-Silva, Johannes C Baayen, Christiaan PJ de Kock, Rhiannon M Meredith, and Huibert D Mansvelder. Mechanisms underlying the rules for associative plasticity at adult human neocortical synapses. *The Journal of Neuroscience*, 33(43):17197–17208, 2013.

[VKLL16] Daniela Vallentin, Georg Kosche, Dina Lipkind, and Michael A Long. Inhibition protects acquired song segments during vocal learning in zebra finches. *Science*, 351(6270):267–271, 2016.

[VP96] Jonathan Victor and Keith P Purpura. Nature and precision of temporal coding in visual cortex: A metric-space analysis. *Journal of Neurophysiology*, 76(2):1310–1326, 1996.

[vR01] Mark CW van Rossum. A novel spike distance. *Neural Computation*, 13(4):751–763, 2001.

[VSZ+11] TP Vogels, Henning Sprekeler, Friedemann Zenke, Claudia Clopath, and Wulfram Gerstner. Inhibitory plasticity balances excitation and inhibition in sensory pathways and memory networks. *Science*, 334(6062):1569–1573, 2011.

[VT01] Rufin Van Rullen and Simon J Thorpe. Rate coding versus temporal order coding: What the retional ganglion cells tell the visual cortex. *Neural Computation*, 13(6):1255–1283, 2001.

[WGNB05] Huai-Xing Wang, Richard C Gerkin, David W Nauen, and Guo-Qiang Bi. Coactivation and timing-dependent integration of synaptic potentiation and depression. *Nature Neuroscience*, 8(2):187–193, 2005.

[XAS14] Mingshan Xue, Bassam V Atallah, and Massimo Scanziani. Equalizing excitation–inhibition ratios across visual cortical neurons. *Nature*, 511:596–600, June 2014.

[XJPD12] Shengjin Xu, Wanchen Jiang, Mu-Ming Poo, and Yang Dan. Activity recall in a visual cortical ensemble. *Nature Neuroscience*, 15(3):449–455, 2012.

[XS04] Xiaohui Xie and H Sebastian Seung. Learning in neural networks by reinforcement of irregular spiking. *Physical Review E*, 69(4):041909, 2004.

[XZZ13] Yan Xu, Xiaoqin Zeng, and Shuiming Zhong. A new supervised learning algorithm for spiking neurons. *Neural computation*, 25(6):1472–1511, 2013.

Universität Bremen
Fachbereich 1
Dr.- rer. - nat.

E r k l ä r u n g

Hiermit versichere ich, dass ich

1. die Arbeit ohne fremde Hilfe angefertigt habe,

2. keine anderen als die von mir angegeben Quellen und Hilfsmittel benutzt habe

und

3. die den benutzten Werken wörtlich oder inhaltlich entnommenen Stellen als solche kenntlich gemacht habe.

_____ ,den _____

_____

(Unterschrift)

Universität Bremen

Datum ----------

Name: _____

Anschrift: _____

Erklärung gemäß § 5 Abs. 2 Nr.3 und § 7 Abs. 1 Nr. 2

.

- Hiermit erkläre ich, dass ich an einer anderen Stelle weder die Annahme als Doktorand noch die Eröffnung eines Promotionsverfahrens beantragt habe.*

- Hiermit erkläre ich, dass ich bereits an einer anderen Stelle die Annahme als Doktorand beantragt habe.*

- Hiermit erkläre ich, dass ich mich bereits einem Promotionsverfahren unterzogen habe.*

* Zutreffendes bitte ankreuzen
(gegebenenfalls Unterlagen beifügen)

Weiterhin erkläre ich, dass ich den Inhalt der Promotionsordnung sowie die „Erläuterungen und Hinweise zur Promotionsordnung“ zur Kenntnis genommen habe. Beide Dokumente sind auf den Internetseiten des Fachbereichs 1 zugänglich und werden auf Wunsch durch das Prüfungsamt des Fachbereichs 1 ausgehändigt.

_____
(Unterschrift)