



Graz University of Technology
Institute for Theoretical Computer Science

Master's Thesis

DESIGN OF A ROBUST WTA-CIRCUIT FOR
THE DISCOVERY OF HIDDEN CAUSES (VIA
STDP) WITH REALISTIC MODELS FOR
EXCITATORY AND INHIBITORY NEURONS

Michael Unger

Graz, Austria, September 2010

Supervisor

o.Univ.-Prof. Dr. Wolfgang Maass

Instructors

Mag. Dr. Stefan Häusler

Dipl.-Ing. Bernhard Nessler

Abstract

Spike-based Expectation Maximization (SEM) gives a powerful tool that combines Spike-Timing Dependent Plasticity (STDP) with Winner-Take-All (WTA) circuits of spiking neurons to learn implicit generative models for high-dimensional spiking input. However, the WTA feedback used is still computed and applied in a biologically impossible way.

The thesis at hand addresses this issue by proposing a spiking feedback loop using realistic neuron models. Computer simulations have been used to evaluate the introduced spiking WTA circuit. Measures for assessing the designed network's classification performance and its ability to learn a representative generative model have been computed and compared to SEM reference values. It turned out that SEM behavior can be approximated for special cases of uniform input distributions.

Throughout the designed WTA circuit, biologically plausible spiking neuron models have been used and parameters were set to be within a biologically reasonable range.

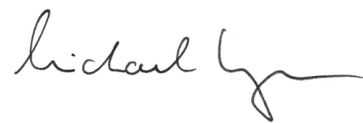
Keywords. WTA-Circuits, STDP, Spiking Neurons, Experimental Analysis, SEM

Acknowledgements

I would like to express my sincere gratitude to Dr. Wolfgang Maass. He provided me with freedom for creativity and the choice of research directions, while at the same time offering guidance based on his profound experience and knowledge. Not only did he find time whenever needed, but his continuous encouragement helped me to progress. I would also like to thank him for enabling me to travel and attend meetings, workshops and conferences. This contributed to extending my perspectives on the research field and introduced me to various interesting people.

Further, I wish to thank my instructors Bernhard Nessler and Stefan Häusler for their support, patience and vital advice which I highly appreciate. Discussions with Bernhard have helped me to focus on and clarify the essentials of my thesis, and not to drift apart. Stefan provided valuable advice with his wide biological and methodological knowledge. He enriched this work through important suggestions and remarks.

Finally, I would like to thank everybody at the Institute for Theoretical Computer Science for a welcoming and inspiring working environment.

A handwritten signature in black ink, appearing to read "Richard Feynman". The signature is fluid and cursive, with a long horizontal stroke at the end.

Contents

1	Introduction	1
1.1	Problem Statement	1
1.2	Outline	2
1.3	Conventions Used in this Document	3
1.3.1	Naming Conventions	3
1.3.2	Drawing Conventions	4
1.3.3	Simulation Conventions	4
1.4	An Introduction to Spiking Neurons	4
1.5	WTA Circuits	5
1.6	STDP	6
2	Related Work	7
2.1	SEM	7
2.1.1	Soft WTA	8
2.1.2	Learning Rule	8
2.2	Neuron Models	9
2.3	Simulation Framework	9
3	Design of a WTA Circuit	11
3.1	From Theoretical to Realistic Spiking Inhibition	11
3.1.1	Output Rate Normalization	11
3.1.2	Biological Constraints	12
3.1.3	Theoretical Inhibition	13
3.1.4	Realistic Spiking Inhibition	14
3.2	Network Architecture	15
3.3	Design of Network Elements	16
3.3.1	Neuron Models	16
3.3.1.1	Firing Rates	17
3.3.1.2	Refractory Period	18

3.3.2	Synapse Models	18
3.3.2.1	Kernel	18
3.3.2.2	Time Constants	20
3.3.2.3	Weights	21
3.3.2.4	Delay	23
3.4	Network Input	23
4	Results	27
4.1	Key Design Choices	27
4.2	WTA Circuit Evaluation Plan	29
4.2.1	Task Definitions	29
4.2.2	Definition of Classification Performance	30
4.2.3	Definition of Generative Model Validation	30
4.3	Inhibition Characteristics	31
4.4	Weight Convergence	32
4.5	Output Spike Evolution	32
4.6	Classification Performance	36
4.6.1	Uniform Pattern Distribution with a z-Neuron Surplus	36
4.6.2	Uniform Pattern Distribution with a z-Neuron Shortage	39
4.6.3	Uniform Pattern Distribution: Number of z-Neurons vs. Number of Patterns	42
4.6.4	Non-Uniform Pattern Distribution	42
4.7	Generative Model Validation	46
4.7.1	Uniform Pattern Distribution without Noise Breaks	46
4.7.2	Uniform Pattern Distribution with Noise Breaks	46
4.7.3	Non-Uniform Pattern Distribution with Noise Breaks	46
4.8	Exponential Inhibitory Neuron	52
5	Conclusion and Future Work	53
5.1	Conclusion	53
5.2	Future work	55
A	Acronyms and Symbols	57
	Bibliography	59

Chapter 1

Introduction

Contents

1.1	Problem Statement	1
1.2	Outline	2
1.3	Conventions Used in this Document	3
1.4	An Introduction to Spiking Neurons	4
1.5	WTA Circuits	5
1.6	STDP	6

1.1 Problem Statement

Recent findings by Nessler, Pfeiffer and Maass [27, 28] have helped to gain new insights how the interplay of Spike-Timing Dependent Plasticity (STDP) and Winner-Take-All (WTA) circuits of spiking neurons can learn implicit generative models for high-dimensional spiking input and detect hidden causes within it. However, the feedback used for WTA competition to emerge is still computed and applied in a biologically impossible way.

This master’s thesis addresses the problem of how to approximate the functionality of the WTA circuit provided in their work by a network of biologically realistic models for excitatory and inhibitory neurons in an experimental way. It is desired to determine scenarios where the proposed network can fit closely to the cited work.

The term WTA circuit (Section 1.5) refers to a specific topology of neurons, that compete by inhibiting each other. While a winning neuron remains active, the others get suppressed. Various kinds of WTA circuits are a common topic of interest in fields related

to computational intelligence and neuroscience. Different implementations with spiking or non-spiking neurons have already been published.

The aim of this thesis is to design a WTA circuit inspired by the theory of Spike-based Expectation Maximization (SEM) [28] by making use of the learning rule proposed therefore. The most important part thereby is the construction of a spiking feedback loop that provides essential normalization inhibition for the WTA competition between the network's output neurons. It has to be analyzed how close the spiking inhibition can fit a theoretically optimal one and how alterations in terms of non-optimal feedback affect the network's performance.

A goal is to keep the designed network reasonably biologically realistic. Especially the spiking behavior of neuron models used should be biologically valid and explainable.

Computer simulations will help to identify scenarios where a good approximation can be made and depict the designed network's limitations. Specific behavior should be observed and analyzed. The primary ambition is to be able to determine benefits and limitations of the proposed completely spiking model. It is desired to assure that comparable performance to the theoretical model can be reached with a biologically realistic one.

In the end, a spiking WTA network of realistic neuron models should be available that can learn to detect hidden causes in high-dimensional input. Results of computer simulations performed with this network should be compared with simulations of the existing model SEM.

1.2 Outline

A short introduction to spiking neurons is given in Section 2.2 as this could help the novice reader to gain a basic idea and help to follow the subsequent work. The concepts of WTA circuits and STDP are predominantly present throughout the whole thesis. Short introductions are given in Section 1.5 and 1.6 to establish a theoretical fundament.

Chapter 2 focuses on recent work that is closely related to the proposed spiking WTA circuit. In Section 2.1, the SEM theory [27, 28] that motivated this thesis is introduced briefly. The soft WTA (Section 2.1.1) as well as the learning rule (Section 2.1.2) are accented as they are especially relevant for subsequent work. Neuron models are discussed in Section 2.2, and their biological plausibility is observed. The chapter is concluded with a specification of the simulation framework used in Section 2.3.

The proposed spiking WTA network is presented in Chapter 3 where also its design is explained. Section 3.1 focuses on how to transfer the theoretical inhibition proposed to

a realistically spiking one. Difficulties and necessary approximations made are explained here. The architecture chosen for the network is visualized in Section 3.2 before Section 3.3 steps through network elements that had to be taken care of while adapting the WTA circuit to the specific needs for this thesis. The choice of parameters is mentioned and argued. It is mentioned when abstractions of biological realism had to be made in favor of simplicity (for comprehensible analyzations) or computational efficiency. Qualitative impacts on simulation results are tried to be estimated. Section 3.4 describes the input used for computer simulations of the proposed network.

In Chapter 4 simulation results are presented and discussed. Data obtained from simulations of the designed model are set in relation to and compared with the theoretical model [28] as a reference. In the following, the designed circuit is analyzed for its classification performance (Section 4.6), focussing on the output spiking behavior and how well they represent specific input patterns. Some variations of input or network configurations are studied, and limitations addressed. A qualitative comparison of the spiking inhibition and the theoretical one is given in Section 4.3. To observe the learning process of the network, the development of its input weights (Section 4.4) as well as its output spikes (Section 4.5) are pictured and discussed throughout a training process beginning with randomly initialized weights and going until their convergence. Observing how far the log-likelihood can be maximized in relation to the optimal model gives an idea of how well an actual generative model representing the input distribution can be learned. This is analyzed in Section 4.7.

Finally, Chapter 5 ends the thesis with a short conclusion in Section 5.1 that recapitulates on findings made throughout this work and ideas that could motivate future research in Section 5.2.

1.3 Conventions Used in this Document

1.3.1 Naming Conventions

To distinguish between the designed WTA model and the SEM model, the following names are used to denote them:

Biological Model This name refers to the WTA model designed within this thesis. Although the designed model is as an approximation far from a completely biologically realistic one, we use this name to refer to the network as it is inspired by biology and communication within the entire network completely based on spikes.

Theoretical Model This name refers to the SEM model (Section 2.1) that serves as a reference implementation the simulation's results are compared to.

1.3.2 Drawing Conventions

Schematic network drawings within this document use the symbols shown in Figure 1.1 to illustrate specific network elements.

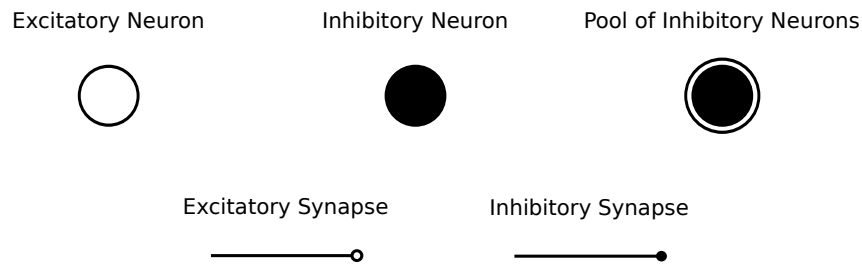


Figure 1.1: All schematic network drawings in this document use the listed symbols for excitatory and inhibitory neurons and connections.

1.3.3 Simulation Conventions

All computer simulations (Section 4) use the default configuration parameters (as not noted otherwise) listed in Table 1.1.

$\# y_n = 100$	The number of input neurons y_n of a network.
$T_{\text{Pattern}} = 50\text{ms}$	Duration of an input pattern.
$T_{\text{Noise}} = 50\text{ms}$	Duration of noise separating the presentation of two input patterns.
$f_{y,\text{Pattern}} = 20\text{Hz}$	Poisson rate of an input neuron y_n while a pattern is active (Section 3.4).
$f_{y,\text{Noise}} = 20\text{Hz}$	Poisson rate of an input neuron y_n when randomly spiking.

Table 1.1: Default network configuration parameters for computer simulations.

1.4 An Introduction to Spiking Neurons

The brain consists of various kinds of cells, that differ slightly from traditional tissue cells. Brain cells can basically be separated into two groups: *neurons* (*nerve cells*) that communicate with each other by emitting electrical pulses and *neuroglial cells*, that work as support cells for the nervous tissue. [30]

There are multiple types of neurons, but they typically consist of a cell body (soma), dendrites and axons. A neuron collects input from other cells via its dendrites and processes the information gained in its soma. Once the input exceeds a specific threshold, a neuron* emits (fires) an electric pulse, called Action Potential (AP), that is forwarded via an axon and transmitted over synapses to other neurons. The amplitude of these electric pulses does not depend on the strength of the input. As these AP's are typically very short ($\approx 1-2\text{ms}$), they are also referred to as *spikes*. As a single AP does not differ from another, the information transmitted between neurons is modulated in the spike occurrence number and time. [12]

An AP is transferred from one neuron to another over a *synapse* by releasing *neurotransmitters*[†] that cause a change of the electrical potential on the postsynaptic neuron. Depending on the synapse type, different kinds of neurotransmitters are set free and cause excitatory or inhibitory postsynaptic potentials on the receiving (postsynaptic) neuron. These are either increasing or decreasing the likelihood of a postsynaptic neuron to fire. [30]

Mathematical models used to describe the dynamics of biological neurons have proven to be of interest for quite a long time now, as the computational power of spiking neurons has been demonstrated [21, 22]. Hodgkin-Huxley-like [16] neuron models provide rich detail down to the ion-channel level, but make analysis in terms of computational abilities hard. Simpler models like the Spike Response Model (SRM) [12] have made a step into being more accessible to analyze while remaining reasonably accurate [19]. [22]

1.5 WTA Circuits

The term WTA circuit refers to a specific network topology composed of typically excitatory pyramidal neurons that inhibit each other through inhibitory interneurons. The resulting competition between the neurons lets a winning neuron fire while the others are suppressed, thereby enhancing the contrast of signals. Recurrent networks are thought to be found in cortex [10].

The computational power of WTA circuits has been demonstrated [23, 24]. They constitute an important role in computational neuroscience and related fields [14, 20, 27, 29, 31, 37].

A hard WTA circuit is a simple deterministic network, where the only winner is selected

*In the following, other types than spiking neurons are neglected.

[†]Synapses emitting neurotransmitters are typically referred to as *chemical synapses*. As these make the major part of synapses for intercellular signaling (among neurons), other types of synapses (i.e. *electrical synapses*) are neglected here.

as the one with the highest input. The winner is active while the others remain silent. A stochastic (or soft) WTA circuit (Section 2.1.1) additionally introduces a probabilistic component. Here, the neuron with the highest input is indeed the most likely, but there remains a probability for other output neurons to fire as well.

1.6 STDP

The amplitude of a postsynaptic potential triggered by a presynaptic spike is determined by the synaptic weight (efficacy) w between them. The ability to change these weights is called synaptic plasticity. If, through specific stimulation, the connection is strengthened over time, it is referred to as Long-Term Potentiation (LTP), if it is weakened, it is called Long-Term Depression (LTD) [12]. A basic mechanism for synaptic plasticity is referred to as Hebbian learning, as Donald O. Hebb stated in his work *The Organization of Behavior* [15]:

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.

Thus, correlation between pre- and postsynaptic activity can induce changes in synaptic weights.

STDP is an extended form of Hebbian learning. The change of a synaptic connection strength is a function of the relative timing of pre- and postsynaptic spikes.[32] It is assumed that STDP plays a major role for learning in biological neural circuits [2]. The function of how the weight changes, with time windows for the pre- and postsynaptic spike arrival of a few milliseconds, differs with various types of neurons and synapses and depends whether the connection is an inhibitory or an excitatory one [5, 7].

Chapter 2

Related Work

Contents

2.1 SEM	7
2.2 Neuron Models	9
2.3 Simulation Framework	9

2.1 SEM

Nessler, Pfeiffer and Maass [27] recently provided a theory of how to approximate the powerful machine learning tool Expectation Maximization (EM) [3, 9] with the use of STDP in a stochastic WTA circuit.

Through a specific STDP learning rule, a generative model of high-dimensional inputs is learned within the collective of synaptic weights of the circuit. After learning, the output neurons fire with a probability depending on how good the current input fits the learned generative model of the circuit. [28]

The learning of this generative model is done using the unsupervised learning principle EM. During its first stage, the expectation step, the network makes an initial guess by randomly selecting a neuron through the soft WTA circuit to fire at the currently applied input. After the selection of a neuron, the choice is evaluated by the modification of the synaptic weights to the active neuron via the STDP learning rule, what would refer to the second EM stage, the maximization step, maximizing the log-likelihood of this internal model for high-dimensional spike inputs learned. As the basic functionality is working with spiking neurons, Nessler, Pfeiffer and Maass named this application of STDP for WTA circuits Spike-based Expectation Maximization (SEM). [28]

2.1.1 Soft WTA

The WTA circuit used in the context of SEM networks is a stochastic or soft WTA circuit where the firing probability for each of the Output Neurons (**z-Neurons**) for a current input \mathbf{y} is given as

$$p(z_k \text{ fires at time } t \mid \mathbf{y}) = \frac{e^{u_k(t)}}{\sum_{l=1}^K e^{u_l(t)}}, \quad (2.1)$$

where $u_k(t)$ is the membrane potential of output neuron z_k at time t . The membrane potentials of the **z-Neurons** are driven by synaptic inputs and computed as

$$u_k(t) = w_{k,0} + \sum_i w_{k,i} \sum_{t_i^{(f)}} \epsilon(t - t_i^{(f)}), \quad (2.2)$$

where $w_{k,0}$ (that could also be referred to as a resting membrane potential) is an offset constant added to the membrane potential to influence its excitability. $w_{k,i}$ denotes the synaptic efficacy from input neuron y_i to the output neuron z_k . $\epsilon(s)$ models the time course of a PostSynaptic Potential (PSP) [12] with $t_i^{(f)}$ marking the times at which the presynaptic input neuron y_i has fired. [28]

2.1.2 Learning Rule

In the maximization step input weights are updated according to the following learning rule:

$$\Delta w_{k,i} = \begin{cases} \eta \left(\sum_{i=0}^t \epsilon(t - t_i^{(f)}) \cdot e^{-w_{k,i}} - 1 \right) & \text{if } z_k \text{ has fired} \\ 0 & \text{else} \end{cases} \quad (2.3)$$

Thereby, η is the current learning rate. If output neuron z_k has fired, its input weights $w_{k,i}$ are updated. If input neuron y_i contributed in exciting z_k

$$\sum_{i=0}^t \epsilon(t - t_i^{(f)}) > 0, \quad (2.4)$$

the synaptic connection inbetween these two neurons $w_{k,i}$ is strengthened, depending on the amount with which the input neuron y_i has increased the membrane potential of output neuron z_k and the current weight $w_{k,i}$.

If however, y_i remained silent some time before output neuron z_k emitted a spike

$$\sum_{i=0}^t \epsilon(t - t_i^{(f)}) = 0, \quad (2.5)$$

and therefore did not contribute in exciting z_k , the connection becomes weaker with

$$\Delta w_{k,i} = -\eta. \quad (2.6)$$

For determining the learning rate η , a variance tracking heuristic, as described in [28], is used for faster convergence of input weights.

2.2 Neuron Models

Detailed conductance-based neuron models, like the one from Hodgkin-Huxley [16], that is detailed to the ion channel level, can capture electrophysiological behavior of neurons in great detail. But due to the complexity of tuning numerous parameters, required for their level of detail, and their high computational demand, they are often not the first choice for large scale simulations. [4]

Integrate-and-Fire neuron models, that filter their inputs and spike once their membrane potential crossed a threshold, are easier to analyze, simulate and fit to real measured data. Due to their simplicity benefits, they are often used for theoretical analysis and computer simulations [12, 25]. It has been demonstrated that such phenomenological models can fit biological data very well [4, 6, 13, 17, 18] and therefore proof to be valid biologically realistic models.

Adding noise to a per definition deterministic Integrate-and-Fire neuron can cause the neuron to fire stochastically [1, 33]. Thus, a neuron emitting inhomogeneous poisson trains with a dependence of the rate on the neuron's input (and therefore membrane potential) can be a plausible abstraction to model biologically realistic neurons.

2.3 Simulation Framework

A simulation framework* for the analysis of learning with WTA circuits was created during my seminar project [35]. Completely written in MATLAB, it offers configurable and highly flexible modules to build, simulate and analyze various kinds of WTA circuits. All simulations done within this thesis were done using this framework.

*Available on request to unger@igi.tugraz.at.

Chapter 3

Design of a WTA Circuit

Contents

3.1	From Theoretical to Realistic Spiking Inhibition	11
3.2	Network Architecture	15
3.3	Design of Network Elements	16
3.4	Network Input	23

3.1 From Theoretical to Realistic Spiking Inhibition

3.1.1 Output Rate Normalization

Competition between output neurons is a vital part in the SEM architecture for learning to emerge (Section 2.1.1). This is achieved by using a soft WTA function (Equation (2.1)) between the z-Neurons. A z-neuron fires according to an inhomogeneous poisson process with an instantaneous rate Λ_k . The output rate of neuron z_k should be proportional to it's membrane potential with an exponential dependence e^{u_k} . Thereby, the sum over the rates of all z-Neurons should remain constant over time $\sum_{k=1}^K \Lambda_k = 1$. Thus, the normalization

$$\Lambda_k = \frac{\lambda_k}{\sum_{l=1}^K \lambda_l} = \frac{e^{u_k(t)}}{\sum_{l=1}^K e^{u_l(t)}} \quad (3.1)$$

arises, which keeps the sum over all output rates $\sum_{k=1}^K \Lambda_k$ constant over time and enhances rate differences between the z-Neurons. The divisive normalization term of the theoretical

SEM model

$$\sum_{l=1}^K e^{u_l(t)} \quad (3.2)$$

is computed at every time step with the known values of the z-Neurons membrane potentials $u_k(t)$.

3.1.2 Biological Constraints

With the aim of designing a biologically plausible WTA circuit where information among neurons is transmitted only by spikes, constraints arise:

- Synaptic inhibition is typically modeled as an additive term.* The normalization of Equation (3.1) can therefore be seen as

$$\Lambda_k = e^{(u_k - \log \sum_{l=1}^K e^{u_l(t)})}, \quad (3.3)$$

where $\log \sum_{l=1}^K e^{u_l(t)}$ describes the required inhibition.

- The membrane potentials $u_k(t)$ for computing the inhibition can not be accessed directly from an inhibitory neuron, as the only information source for a neuron are its input spikes. Therefore, only the effective presynaptic membrane potentials v_k for computing the inhibition can be estimated from the z-Neurons output spikes.
- An estimate for the neurons membrane potentials has to be sampled from output spikes of the z-Neurons. As the z-Neurons within a spiking WTA circuit receive inhibitory feedback, their membrane potentials are affected by the Inhibition (\mathcal{I}) and are therefore computed as $v_k(t) = u_k(t) - \mathcal{I}(t)$. Thus, the output of a z-neuron does not only depend on the part $u_k(t)$ of the membrane potential, driven by input neurons y_i , but on the complete membrane potential v_k including inhibition. With the output spikes of the z-Neurons as the only source of information, the inhibitory neuron can only make an estimate for the effective membrane potential v_k .
- The transmission of a biological AP takes time as it is propagated along a neuron's axon and transmitted over a chemical synapse. This has to be accounted for by introducing a delay in the designed model. In contrast to a SEM network, an instantly applied, correct inhibition is not possible.

*Although models like shunting inhibition exist, additive inhibition is a well established method how synaptic inhibition can be modeled.

3.1.3 Theoretical Inhibition

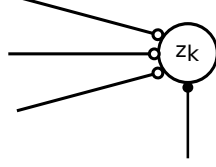


Figure 3.1: A single neuron z_k with excitatory and inhibitory inputs. Its membrane potential u_k is computed in Equation (3.4).

Considering a neuron z_k (Figure 3.1), whose membrane potential is driven by excitatory inputs and an additional inhibitory connection, its membrane potential v_k can be computed analogous to Equation (2.2) with the neurons resting potential named u_0 as

$$v_k(t) = u_k(t) - \mathcal{I}(t) = u_0 + \sum_i w_{k,i} \sum_{t_i^{(f)}} \epsilon_0(t - t_i^{(f)}) - \mathcal{I}(t). \quad (3.4)$$

$\mathcal{I}(t)$ denotes the incoming inhibition at time t . Assuming z_k fires a poisson train with a rate exponentially depending on its membrane potential $v_k(t)$ (Section 3.3.1), the output rate can be given as

$$\Lambda_k(t) = e^{v_k(t)} = e^{u_k(t) - \mathcal{I}(t)}. \quad (3.5)$$

Rewriting the SEM models soft WTA computation (Equation (2.1)) as

$$p(z_k \text{ fires at time } t \mid \mathbf{y}) = \frac{e^{u_k(t)}}{\sum_{l=1}^K e^{u_l(t)}} = \exp \left(u_k(t) - \underbrace{\log \sum_{l=1}^K e^{u_l(t)}}_{\mathcal{I}_{\text{Theoretical}}(t)} \right), \quad (3.6)$$

and given the possibility to obtain the real values of $u_l(t)$ instantaneously, the subtractive inhibition term $\mathcal{I}(t)$ would model the required normalization exactly:

$$\mathcal{I}_{\text{Theoretical}}(t) = \log \sum_{l=1}^K e^{u_l(t)} \quad (3.7)$$

3.1.4 Realistic Spiking Inhibition

In a realistic setup, u_k is not directly accessible. Through the output spikes of neuron z_k (Equation (3.5)), a Low-Pass Filtered (LPF) signal in dependence of its membrane potential v_k can be estimated. The LPF is determined by the time constant τ of the respective EPSP (Section 3.3.2.2). To close the complete spiking, regularizing feedback loop (Figure 3.2) an inhibitory neuron is introduced.

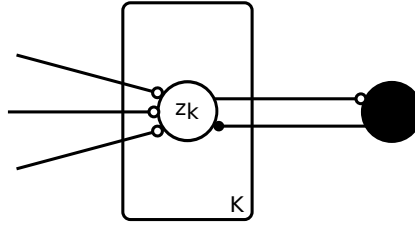


Figure 3.2: Spiking feedback loop. Neuron z_k has an excitatory connection to the inhibitory neuron. This projects back with an inhibitory connection. For the drawing *plate notation* [3] was used. Only one neuron z_k out of K , that all share the same inhibitory neuron, is drawn.

With inputs from all z-Neurons, the inhibitory neuron's membrane potential u_{Inh} is proportional to the sum of LPF estimates of $e^{v_k(t - \Delta^{\text{ax}})}$, where Δ^{ax} is an axonal spike transmission delay:

$$u_{\text{Inh}} = u_0 + \sum_{k=1}^K \text{LPF} \left(e^{v_k(t - \Delta^{\text{ax}})} \right) \quad (3.8)$$

To make a good approximation of the theoretical SEM inhibition (Equation (3.8)), the inhibitory neuron is chosen to be a linear poisson neuron (Section 3.3.1) assuming the output firing rate linearly proportional to its membrane potential u_{Inh} . This is done by linearizing the desired logarithmic dependence around an operating point where the logarithmic function $y = \log(x)$ can be approximately fitted (Figure 3.3) with $y = x - c$. The constant c can be dropped in a neuron's excitability factor u_0 .

By projecting back to the z-Neurons with an inhibitory connection, the spiking feedback loop is closed. Each of the z-Neurons is therefore inhibited by

$$\mathcal{I}_{\text{Spiking}}(t + \Delta^{\text{ax}}) \propto \text{LPF} \left(u_0 + \sum_{k=1}^K \text{LPF} \left(e^{v_k(t - \Delta^{\text{ax}})} \right) \right). \quad (3.9)$$

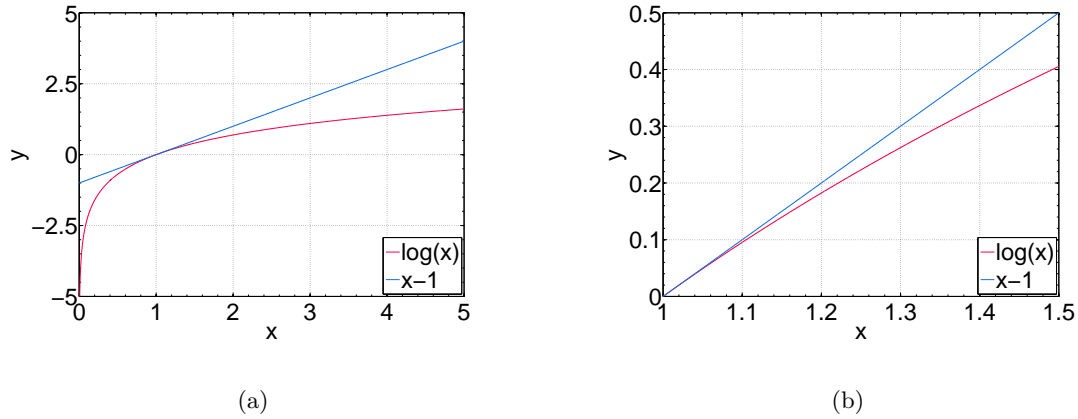


Figure 3.3: Comparing the functions $y_1 = \log x$ and $y_2 = x - 1$. For values of x close to 1, $\log x$ can be approximately linearized.

3.2 Network Architecture

Having an eligible spiking approximation of the theoretical inhibition (Section 3.1.4), a complete WTA circuit can be constructed. A schematic drawing of the network architecture can be seen in Figure 3.4.

The input layer consists of N input neurons y_n (Section 3.4) that have an all-to-all connection to the network's output layer of z -Neurons. Initially weighted randomly, the strengths of these connections are learned using the SEM learning rule (Section 2.1.2). The weight of the connection from an input neuron y_n to an output neuron z_k is denoted as $w_{k,n}$.

The output (z) layer consists of K exponential neurons z_k that have bidirectional connections to an inhibitory layer, so that WTA competition between the z -Neurons can emerge. Connections in both directions are weighted with $w = w_{Inh,Exc} = w_{Exc,Inh} = 1$. These values do not necessarily have to be 1, but should remain equal and constant. Inhibition strength, and therefore the inhibition of z -Neurons can be scaled with these factors. Thus a single spike emitted from any z_k has the same impact on an inhibitory neuron's membrane potential and vice versa, an inhibitory spike affects each of the z -Neurons equally.

To match the approximated SEM inhibition the best, linear neurons are used in the inhibitory pool for further analysis and computer simulations (Section 4). This simplifies interpretations of simulations and makes analysis easier. For a step towards biological realism, simulations with exponentially firing neurons have also been performed (Section 4.8)

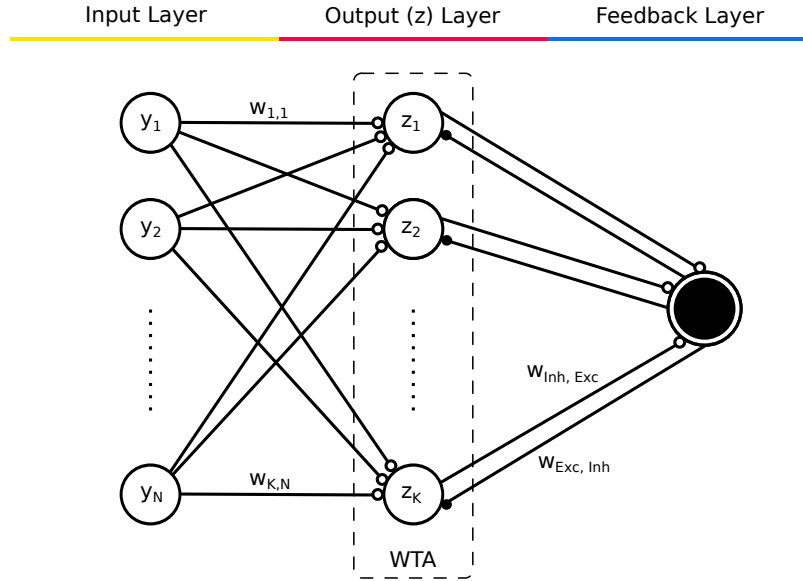


Figure 3.4: The architecture of the designed, biologically inspired network. Additional to a classical SEM setup, the necessary feedback for WTA competition at the z-Neurons is designed with completely spiking information transmission. A pool of inhibitory neurons is used to provide the required inhibition.

and have shown to be qualitatively equal. Despite that, linear neurons have been used for the majority of simulations and analysis as a matter of simplicity.

3.3 Design of Network Elements

3.3.1 Neuron Models

For constructing the network, two different types of poisson neurons have been used:

- Exponential Neurons
- Linear Neurons

Both neurons modulate the rate of a poisson train emitted by a neuron in dependence to a function of their membrane potentials (Section 2.2), and therefore in dependence on their inputs [1].

The output layer of z-Neurons consists of exponential neurons. Their firing rate depends exponentially on their membrane potentials u_k , what seems to be a biologically

suitable assumption [18]. The basic aspect of this firing behavior is summarized in modeling a neuron's poisson firing rate with

$$f_{k,\text{Exp}} = e^{u_k} \cdot f_{\text{Base}}, \quad (3.10)$$

where f_{Base} is a base rate defining the neuron's spike rate scale. The use of exponential neuron models as z-Neurons is already intended in the original SEM theory.

The inhibitory pool is modeled with linear neurons (Section 3.1.4). The firing rate of these neurons has a linear dependence on their membrane potentials u_k . For simulation purpose, they have been modeled with

$$f_{k,\text{Exp}} = u_k \cdot f_{\text{Base}}. \quad (3.11)$$

3.3.1.1 Firing Rates

An abstraction in terms of biological feasibility is the choice of the neuron firing rate magnitude for simulation. Despite firing rates of cells in cortex of around 40 – 100Hz [1, 26, 36], simulations (Section 4) were done with base firing rates of $f_{\text{Base}} = 400\text{Hz}$. These base rates are multiplied with the specific neuron's membrane potential $u_k(t)$ for linear neurons or the exponential function $e^{u_k(t)}$ for exponential neurons (Section 2.2) to calculate the effective neuron poisson firing rate $f_k(t)$.

Although those rates exceed biological rates by far, simulation results are not qualitatively different. As increased firing rates make a huge impact on computer simulation time, the effect of a reduced number of weight updates inbetween two spike events can be made up by choosing an appropriate learning rate and adapted simulation environment. Hence the frequency shift out of a biologically realistic region is a technical simplification only and does not imply a qualitative biological violation.

In a further simulation abstraction, a single neuron x_{Inh} was used to model the pool of originally I inhibitory neurons. Again, this has no qualitative impact on the simulation as every z-neuron has equal connections to every neuron in the inhibitory pool [26], and vice versa. Thus, having a single inhibitory neuron spiking with

$$f_{x_{\text{Inh}}} = \sum_{i=1}^I f_{x_i}, \quad (3.12)$$

is an acceptable simplification to gain a benefit in simulation computation time. Furthermore, by distributing the required frequency band of x_{Inh} over a pool of multiple inhibitory

neurons, the firing rate of a single neuron could be easily shifted into a biologically plausible region.

3.3.1.2 Refractory Period

A neuron's refractoriness is described by its Absolute Refractory Period (ARP) and Relative Refractory Period (RRP). During a specific time following an AP, called the ARP, it is impossible to trigger a second spike from the same neuron. During the RRP it becomes possible to initiate an AP again, but it is inhibited. [12]

While a RRP was not modeled for reasons of analyzing simplicity, an ARP is included through the simulation timestep Δt , as only one AP can be generated per Δt . With a simulation resolution of $\Delta t = 1\text{ms}$, we define a maximum firing rate f of a single neuron with

$$f_{k,\max}(t) = \frac{1}{\Delta t} = \frac{1}{\Delta^{\text{ARP}}} = \frac{1}{1\text{ms}} = 1000\text{Hz}. \quad (3.13)$$

The length of an ARP determines a limit of how far the output rate f_k can manage to stay proportional to the neuron's membrane potential v_k . Figure 3.5 demonstrates this effect for the postsynaptic output of a linear neuron. As long as the neuron's output rate is below its maximum value $f_k(t) < f_{k,\max}$, its postsynaptic output stays proportional to the membrane potential, but saturates once $f_{k,\max}$ is reached.

3.3.2 Synapse Models

A presynaptic spike at time $t_i^{(f)}$ increases – Excitatory PostSynaptic Potential (EPSP)– or decreases – Inhibitory PostSynaptic Potential (IPSP)– a postsynaptic neuron's membrane potential v_j for $t > t_i^{(t)}$ by an amount of

$$w_{j,i} \cdot \epsilon_{j,i}(t - t_i^{(f)}). \quad (3.14)$$

The time course of the PSP is described by a kernel $\epsilon(s)$ (Section 3.3.2.1). $w_{i,j}$ denotes the synaptic efficacy (Section 3.3.2.3). To model the transmission of an AP, a delay can be included in the kernel $\epsilon(s)$ (Section 3.3.2.4). [11]

3.3.2.1 Kernel

Every synaptic connection within the network uses the same basic kernel $\epsilon(s)$ to model a PSP, although they differ in the choice of parameters. A kernel containing an exponential

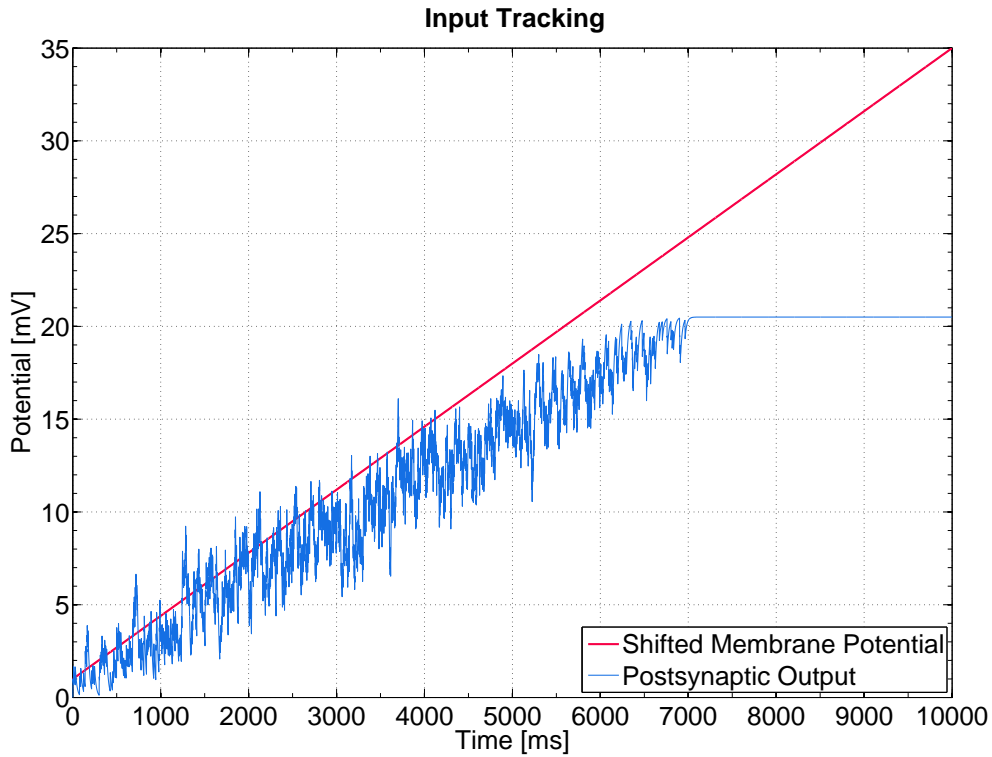


Figure 3.5: A linearly increasing membrane potential compared to its postsynaptic output. Due to a neuron’s refractory period, it can not fire infinitely fast and thus, can not follow the increasing membrane potential once the neuron’s maximum rate is reached.

decay

$$\epsilon(s) = \exp\left(-\frac{s}{\tau}\right)\Theta(s) \quad (3.15)$$

was used. τ is the the time constant of the decaying exponential function (Section 3.3.2.2). $\Theta(s)$ denotes the Heaviside step function with $\Theta(s) = 1$ for $s > 0$ and $\Theta(s) = 0$ else. [12]

Figure 3.6 shows the time course of the kernel $\epsilon(s)$ for two different time constants τ used within the designed network. As a spike occurs at $t_i^{(f)} = 10\text{ms}$, the PSP rises instantly. This assumption might not be biologically exact, but holds as an appropriate approximation. As biological findings show that rise times of postynaptic potentials can be less or around $\Delta t_{\text{Rise}} = 1\text{ms}$ [34], it is assumed smaller than the simulation time resolution. Additionally, PSP rise times, determined by an exponential rising function

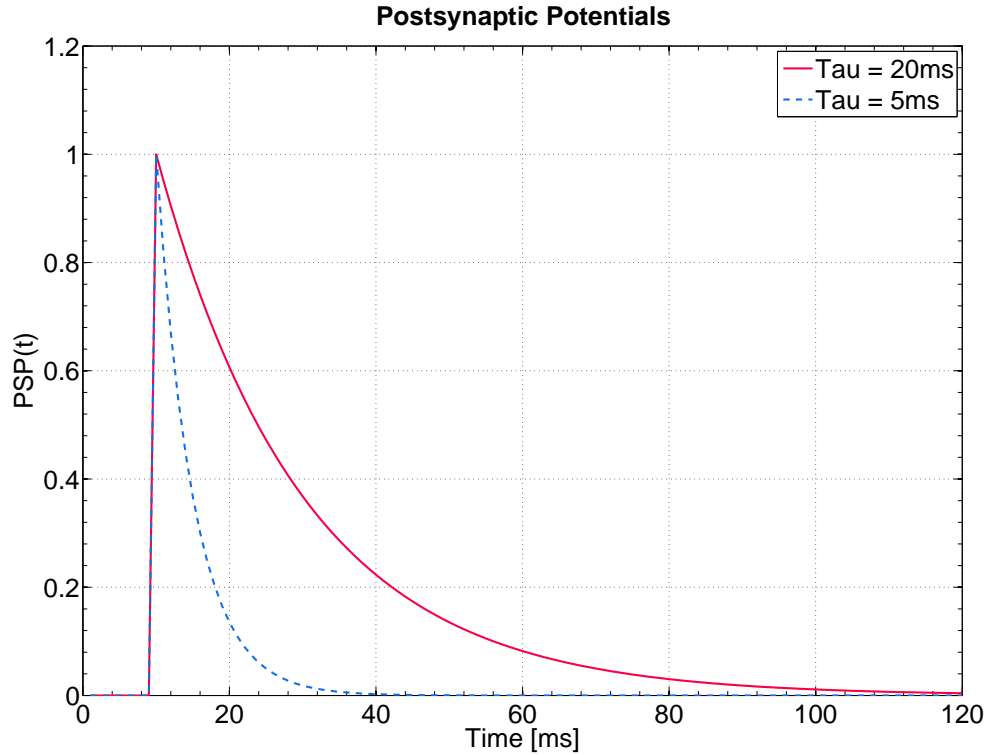


Figure 3.6: Unweighted kernel $\epsilon(t - t_i^{(f)})$ for the modelling of postsynaptic potentials with different time constants τ over time t . The only spike occurred at $t_i^{(f)} = 10\text{ms}$.

with time constant τ_m , are very short compared to their fall times defined by $\tau_s = \tau$,

$$\tau_m \ll \tau_s, \quad (3.16)$$

so that this abstraction was done in favor of computational effort and simplicity.

3.3.2.2 Time Constants

The time constant τ determines the width of the exponential kernel $\epsilon(s)$ (Figure 3.6) convolved with a presynaptic AP to compute a LPF signal of the original spike train. This stands as the effect on the postsynaptic membrane potential.

The values used for simulating the network were set to be in a biologically plausible region [34]. Their exact value was determined by empirical simulation. The aim was to have a neuron's LPF output signal follow the desired output characteristic in relation to

the membrane potential. Figure 3.7 shows an example, performed on a linear neuron. Therefore, the neurons membrane potential v_i should be followed qualitatively by the convolved output signal. A characteristic membrane potential v_i is applied to trigger the presynaptic neuron.

Figure 3.7(b) shows an extreme case of a too short time constant τ_b . Here, the postsynaptic neuron j is not able to track the presynaptic membrane potential v_i , because information is only available for a short period of time. Figure 3.7(a) demonstrates an opposite example with a τ_c too large. As a PSP is too long information about previous spikes builds up and the correlation is low again. Figure 3.7(a) finally shows a proper choice for the time constant τ_a with a correlation coefficient of $\text{corr}_a = 0.831$.

For the construction of the network two time constants have been used:

- $\tau_1 = 20\text{ms}$ – For connections from input neurons y_n to z-Neurons.
- $\tau_2 = 5\text{ms}$ – For connections from z-Neurons to the inhibitory pool and back.

3.3.2.3 Weights

As the shape of how a presynaptic AP at $t_j^{(f)}$ affects the postsynaptic membrane potential v_i is determined by the kernel $\epsilon(s)$, its amplitude is determined by the strength of the synaptic connection $w_{i,j}$. By adding a sign to the synaptic efficacy $w_{i,j}$, a distinction between an EPSP and an IPSP can be made. An EPSP typically causes the membrane potential to depolarize and the neuron to fire more likely. An IPSP causes hyperpolarization, which inhibits the neuron's firing. [11]

The input layer weights are initialized randomly (with a mean magnitude of 0.005) and learned with the SEM learning rule (Section 2.1.2). They are set to be in a region to cause plausible PSP values with a magnitude of approximately $0.1 - 3\text{mV}$ [34].

Weights $w_{\text{Inh, Exc}}$ from the z-Neurons to the inhibitory pool are all equal. This accounts for the same impact from any of the z-Neurons on the inhibition. Equivalently, the weights from the inhibitory pool back to the z-Neurons $w_{\text{Exc, Inh}}$ are also identical to provide the same inhibition for each of the z-Neurons. Again, this seems to be a valid assumption as weights from an inhibitory interneuron onto different pyramidal cells are not necessarily of equal strength, but seem to change accordingly [26]. As we do not learn these weights, the assumption can be argued because the relation within the different synapses stays the same. By having the same excitability on all z-Neurons these weights are chosen equally. The abstraction made by summarizing the inhibitory pool with a single neuron for simulation

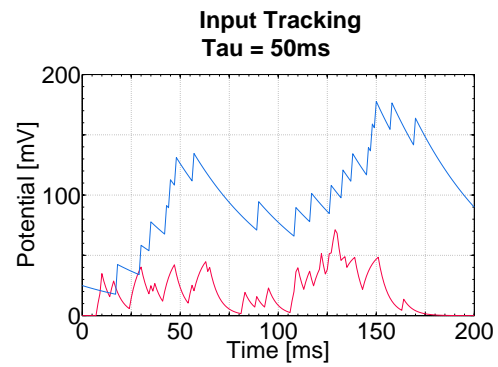
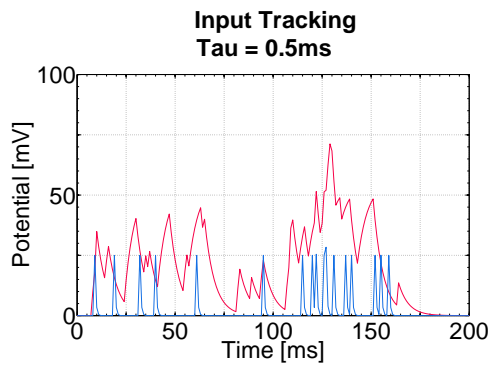
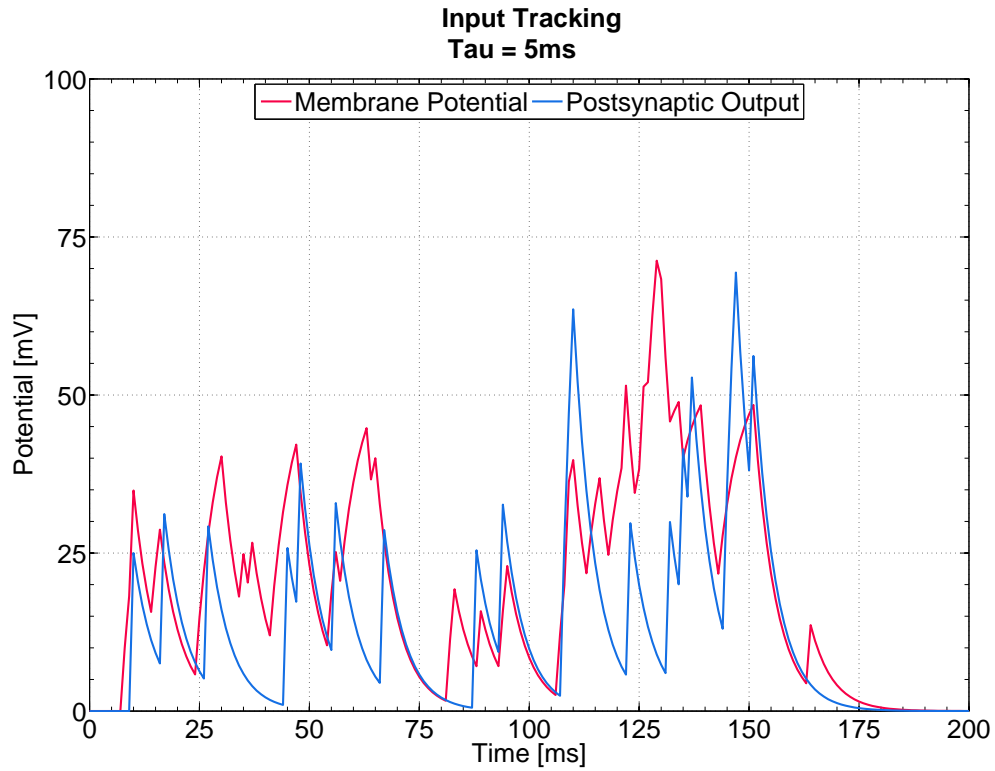


Figure 3.7: The correlation between a linear neuron's membrane potential v_i and its spiking output convolved with an exponential kernel $\epsilon(s)$ with various time constants τ . (a) $\tau_a = 5\text{ms}$, $\text{corr}_a = 0.831$ (b) $\tau_b = 0.5\text{ms}$, $\text{corr}_b = 0.403$ (c) $\tau_c = 50\text{ms}$, $\text{corr}_c = 0.363$

has to be taken into account when choosing the weights $w_{\text{Inh, Exc}}$ and $w_{\text{Exc, Inh}}$. Therefore, the weights are set to 1. The offset caused by high weight values can be made up for by adapting the neurons resting potential.

3.3.2.4 Delay

Information transmission over synapses does not happen instantly. An axonal transmission delay Δ^{ax} (Figure 3.8) is added to the kernel $\epsilon(s)$ [11]:

$$\epsilon(s) = \exp\left(-\frac{s - \Delta^{\text{ax}}}{\tau}\right)\Theta(s - \Delta^{\text{ax}}) \quad (3.17)$$

Simulations have shown that values of a complete feedback loop delay

$$\Delta_{\text{Inh, Exc}}^{\text{ax}} + \Delta_{\text{Exc, Inh}}^{\text{ax}} > 4\text{ms} \quad (3.18)$$

lead to poor network performance. Therefore, values for these delays are set to $\Delta^{\text{ax}} = 1\text{ms}$ for every synaptic connection within the network [8].

3.4 Network Input

An input sequence to the network is a sequence of predefined patterns alternating with noise. Every pattern Pat_l presented is chosen randomly out of L different patterns and followed by a sequence of T_{Noise} with noise.

A pattern Pat_l is generated once by generating a random poisson spike train with rate f_{Pat} and duration T_{Pat} for every single neuron y_n of N input neurons. The exact spike trains are stored and applied as input to the network every time the pattern Pat_l is chosen and presented.

After each pattern, every neuron y_n of N input neurons fires a randomly generated poisson train as a noise sequence with the same rate $f_{\text{Noise}} = f_{\text{Pat}}$ and duration T_{Noise} .

For simulations carried out (Section 4), a poisson rate of $f_{\text{Noise}} = f_{\text{Pat}} = 20\text{Hz}$ and a pattern duration $T_{\text{Pat}} = 50\text{ms}$ was used. Simulations were performed with noise between patterns of duration $T_{\text{Noise}} = T_{\text{Pat}} = 50\text{ms}$ or $T_{\text{noise}} = 0\text{ms}$ as noted.

A sample input sequence consisting of $L = 3$ different patterns of duration $T_{\text{Input}} = 400\text{ms}$ is presented in Figure 3.9. Note that the spike trains are exactly the same as a pattern is repeatedly presented.

Further simulations where pattern sequences and noise interfere have been done.

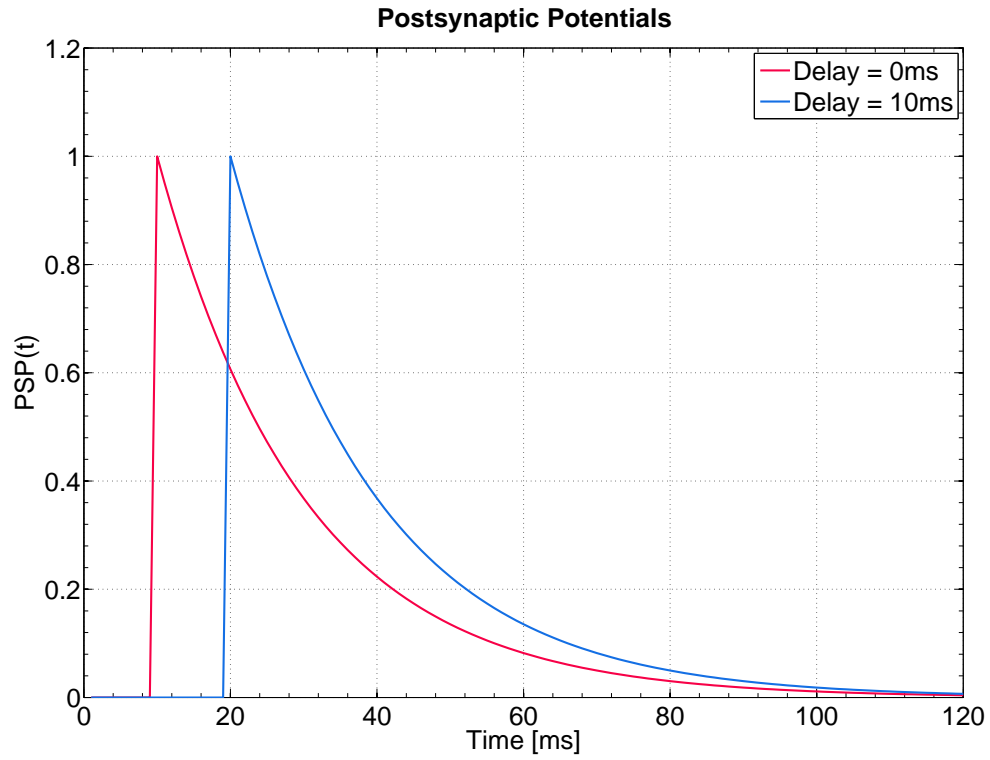


Figure 3.8: Axonal transmission delay Δ^{ax} shifting the arrival of a postsynaptic potential back.

Therefore, the patterns were created with a rate of $f_{\text{Pat}} - f_{\text{superimposed Noise}}$ and superimposed by a poisson noise train with a rate of $f_{\text{superimposed Noise}}$ every time they occurred as a network input. This had no qualitative impact, but weakened and blurred the performance of simulations. For demonstration clarity, noise-superimposed patterns have been neglected in the following.

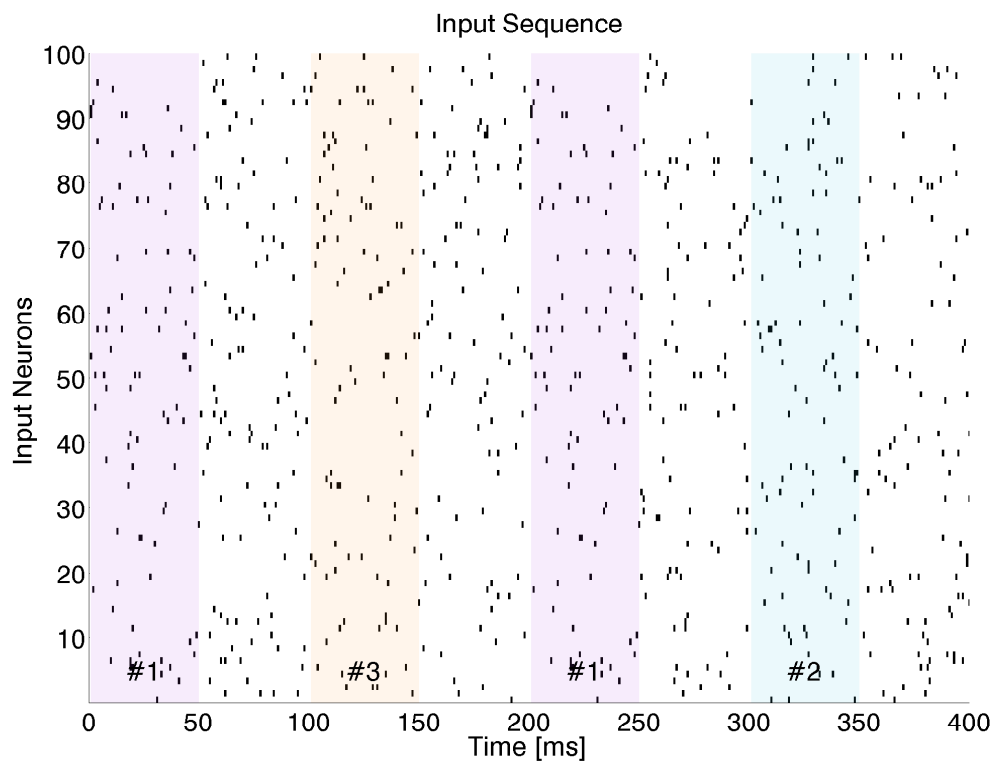


Figure 3.9: Sample of an input spike sequence to a network consisting of $L = 3$ different patterns of duration $T_{\text{Input}} = 400\text{ms}$ with $f_{\text{Noise}} = f_{\text{Pat}} = 20\text{Hz}$ and $T_{\text{Noise}} = T_{\text{Pat}} = 50\text{ms}$. Patterns presented are numbered and marked with a specific color while they are active.

Chapter 4

Results

Contents

4.1	Key Design Choices	27
4.2	WTA Circuit Evaluation Plan	29
4.3	Inhibition Characteristics	31
4.4	Weight Convergence	32
4.5	Output Spike Evolution	32
4.6	Classification Performance	36
4.7	Generative Model Validation	46
4.8	Exponential Inhibitory Neuron	52

4.1 Key Design Choices

During an iterative design and evaluation process, the choices of a few specific network parameters have shown to be of particular significance. Only with suitable configurations of these parameters, simulations turned out to performe reasonably well:

- **Axonal transmission delay Δ^{ax} :** The choice of the axonal transmission delay for connections to and from the inhibitory neuron turned out to be quite limited. An upper bound for the complete axonal feedback loop delay of

$$\Delta^{\text{Loop}} \leq 4\text{ms} \tag{4.1}$$

appeared to be the experimental limit. The distribution between connections to and from the inhibitory neuron proved to be of less significance. The axonal transmission

delay finally defined for simulations was

$$\Delta^{\text{ax}} = 1\text{ms} \quad (4.2)$$

for both directions. In terms of biological plausibility, this seems to be a reasonable choice [8].

- **PSP time constants τ :** How good a presynaptic membrane potential can be traced by a postsynaptic one, is limited by the time constant τ of the postsynaptic potentials and the presynaptic neuron's firing rate. Tuning the decaying time constant τ_2 for the postsynaptic potentials used for connections within the inhibitory feedback loop, turned out to be vital. In case the filtering time constant was chosen too small or too large the inhibitory neuron could not follow the z-neuron's membrane potential exactly enough. The chosen value of

$$\tau_2 = 5\text{ms} \quad (4.3)$$

was determined experimentally by maximizing the correlation between pre- and post-synaptic potentials.

- **Absolute refractory period (ARP):** The absolute refractory period limits the maximum firing frequency of a neuron. Together with the PSP time constant τ , it plays a major role for estimating presynaptic activity through spikes. For simulations, an absolute refractory period of

$$\text{ARP} = 1\text{ms} \quad (4.4)$$

was chosen. The ARP seems to be a key parameter for the designed network to work in general. We assume that the ARP limits the rate of which the weights change during an initial training phase, and therefore prevents them from drifting apart. This assumption was substantiated by simulations with an $\text{ARP} < 1\text{ms}$.

- **Membrane resting potential u_0 :** The choice of z-Neurons' membrane potentials has to be set in relation with the number of z-Neurons available and therefore, with the inhibition's strength. The more z-Neurons are available, the more spikes are collected initially by the inhibitory neuron and thus, inhibition becomes stronger. As synaptic weights within the feedback loop are changed, the resting potentials

have to be adjusted. Depending on the network setup, a membrane resting potential has to be chosen to keep the z-Neurons spiking though inhibition.

- **Input distribution:** The choice of the input distribution had quite an impact on the generative model learned. With uniformly distributed input patterns, the proposed network seemd to learn a representative generative model of the input distribution. Although classification was still possible, this was not the case for patterns with very unequal occurance probabilities (Section 4.7).

4.2 WTA Circuit Evaluation Plan

In the following, guidelines and test case scenarios of how to assess the performance and the learning abilities of the designed WTA circuit are thought of. For the computer simulations done, the proposed network was implemented in MATLAB (Section 2.3). An additional implementation of a standard SEM model served as reference for comparison.

4.2.1 Task Definitions

For simulations, input as described in Section 3.4 was used. Patterns with poisson trains of constant rate were generated and presented. The number of available input patterns was constant and was not changed during the training and testing phase of a respective simulation run. Two different scenarios of different complexity have been tested. In the basic setup, all available patterns occurred on the input with equal probability (*Uniform Pattern Distribution*). In the more advanced setup, patterns got a prior probability assigned. Therefore, some of them occurred less likely than others (*Non-Uniform Pattern Distribution*).

Before stepping through the given test case scenarios, a look on the inhibition of the designed spiking loop will be taken at first to give an idea of the qualitative differences to SEM inhibition (Section 4.3). While any kind of performance or evaluation measure for the learning ability of the proposed network is meaningless unless its weights have converged and settled around a value, we will see whether input weights of the proposed model can converge at all (Section 4.4). What happens at the output of the z-Neurons during the time it takes for the input weights to converge (the learning phase), is observed in Section 4.5.

For evaluation of the proposed network, two different criteria have been assessed. The network's ability to classify patterns and its generative model learned. Each of them tested

for the two pattern distribution scenarios defined.

4.2.2 Definition of Classification Performance

In order to introduce a method to measure and compare a network's ability to distinguish between different input patterns, the classification performance is introduced. Therefore, spikes within a pattern time window are summed up, thresholded and correlated with a target vector. The resulting measure gives an idea of how good a single output neuron reacts to its most likely target pattern.

The ability of a network to discretize between different input patterns is basically limited by the number of available z-Neurons, as an active output neuron marks an active input pattern. Therefore, the performance was analyzed for configurations with

- Number of z-Neurons \geq Number of Patterns
- Number of z-Neurons $<$ Number of Patterns

4.2.3 Definition of Generative Model Validation

While only having a look at the output performance of a network, it is hard to infer how good an actual generative model learned represents the applied input distribution. Therefore, a look on the generative model learned, represented in the collective of input weights, is more fruitful. Typically assessed by the minimized Kullback-Leibler divergence, that measure did not turn out to be suitable, as a required normalization term could not be found. But maximizing the log-likelihood ($\ell\ell$) can be seen as an equal measure [27], although not in absolute values. As it is shown that SEM learns a valid generative model of the input distribution, it is taken as a reference that the $\ell\ell$ computed for the proposed WTA circuit is compared to:

$$\ell\ell = \sum_{i=1}^S \log \sum_{k=1}^K e^{u_k(i)} \quad (4.5)$$

With maximizing the $\ell\ell$ in the SEM training phase [28], a comparison of how close the designed WTA circuit's generative model $\ell\ell$ can give an idea of how good the network learns.

A lower bound for the $\ell\ell$, with a better learned generative model further above this bound, had to be defined. For that reason, the network the $\ell\ell$ was originally computed for, was taken and its inputs randomly permuted. This was done several times and is

drawn with dotted lines in the corresponding plots. Once the ll goes below this randomly connected value of the reference, the model learned is defined as invalid.

Again, in order to make an assertion about how good a generative model was learned, the learning itself has to be finished. This has happened, once the input weights converged and do not qualitatively change anymore. Simulations were done for both defined input distribution scenarios.

4.3 Inhibition Characteristics

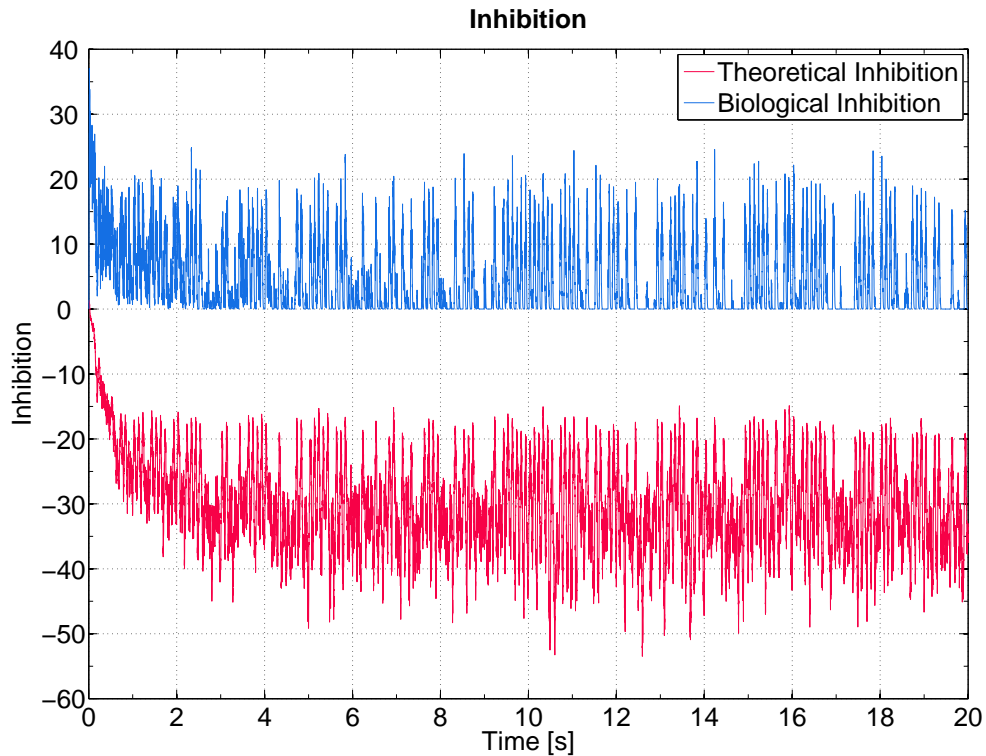


Figure 4.1: Comparison of inhibitions of training a theoretical and a biological model. The curves have a correlation coefficient of $\text{corr} = 0.8420$.

In contrast to theoretical inhibition, spiking inhibition is band limited. It can not react infinitely fast, as the spiking inhibition control loop has to react to the change of its reference signal, the output rate of the z-Neurons. Not only is the spiking inhibition band limited in frequency changes (as the z-Neurons output is LPF), sampled and therefore only estimated, it is also delayed (by axonal transmission delays) and limited in its amplitude.

As a single inhibitory neuron can only emit a single spike at a time and is muted for the following refractory period (Section 3.3.1.2), the maximum inhibition amplitude is limited at every timestep t . The minimal inhibition that can be achieved with this network setup is $\text{Inhibition} = 0$. This is because a neuron can not invert its signals sign, but only remain silent.

Theoretical inhibition on the other hand can apply a optimal computed value immediately at timestep t , without any limitations in its amplitude, rise or fall times, delays or signs.

An example comparison of inhibition signals from a theoretical and a biological network model can be seen in Figure 4.1. The signals have been recorded during training of the networks. The signal from the biological network is printed as the direct spiking output of the inhibitory neuron convolved with an IPSP. The negative sign, characterizing the inhibitory signal, has been left out to make the two signals distinguishable. The mean value of both signals decays at the beginning of the learning phase and settles afterwards. After this transient phase, inhibition settles around a constant mean value.

4.4 Weight Convergence

Input weights, the synaptic strengths from the input layer neurons y_n to the output layer neurons z_k , are learned during the network's training phase. A vital part of the network's performance is to have the input weights converging for a defined set of input patterns, as a network would forget the once learned patterns otherwise.

Figure 4.2 shows an example plot of $K = 4$ z-Neurons input weights evolution while training. At the beginning the weights change, as new patterns are introduced and the learning rule still has effect on the weights. After a while, the weights basically converge and just fluctuate around this value. From this stage on, when the weights have settled, no further learning occurs if the input distribution remains unchanged.

4.5 Output Spike Evolution

To give an idea of how the classification performance from the z-Neurons emerges, Figure 4.3 and Figure 4.4 show the evolution of z-Neurons output spikes during learning. The network with $K = 3$ z-Neurons was trained with $L = 3$ equally likely input patterns (a repeated set of 12 pattern occurrences).

Figure 4.3 shows the beginning of the training sequence. As input is presented, and the

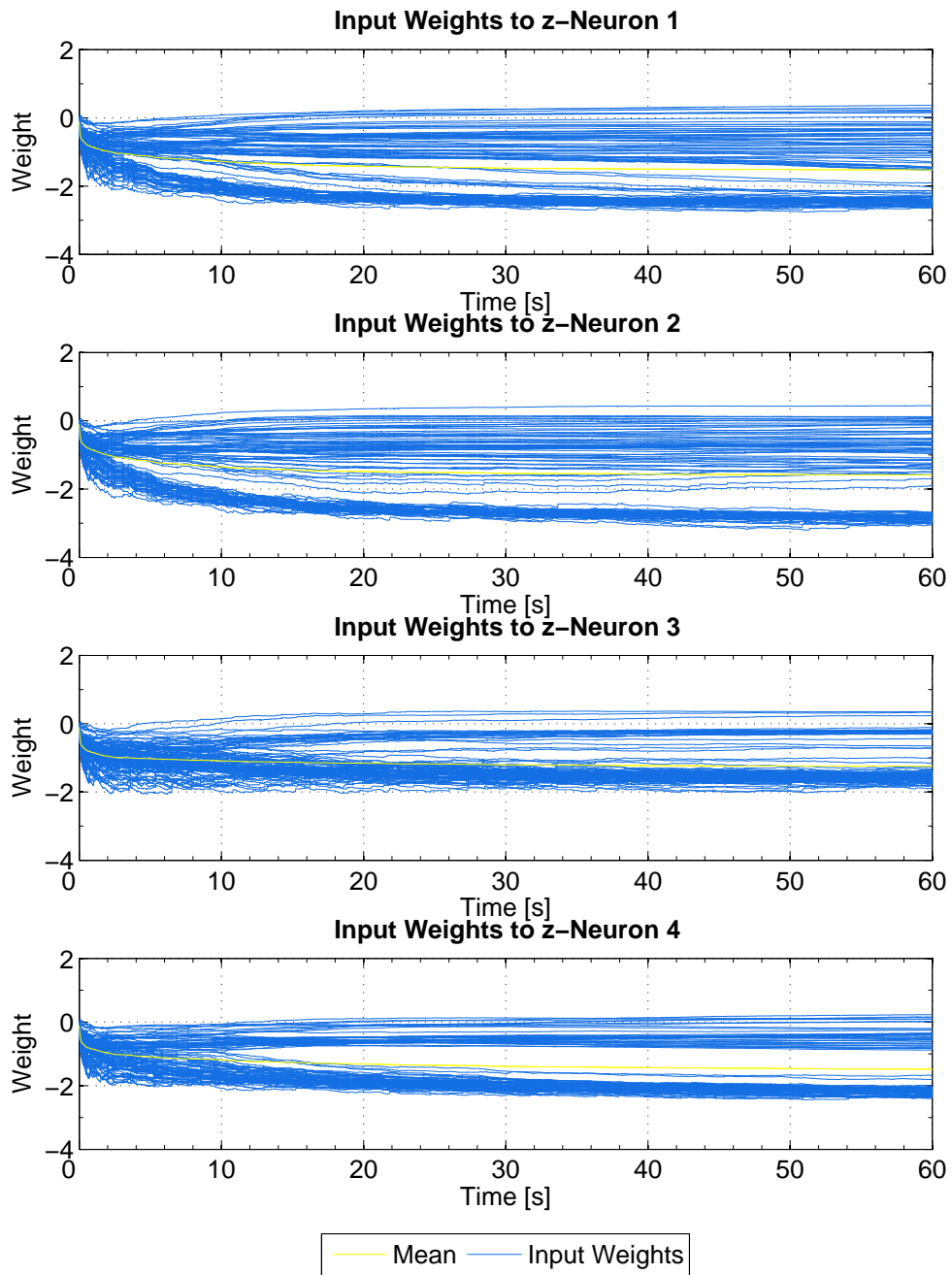


Figure 4.2: Evolution of the input weights to a networks output neurons.

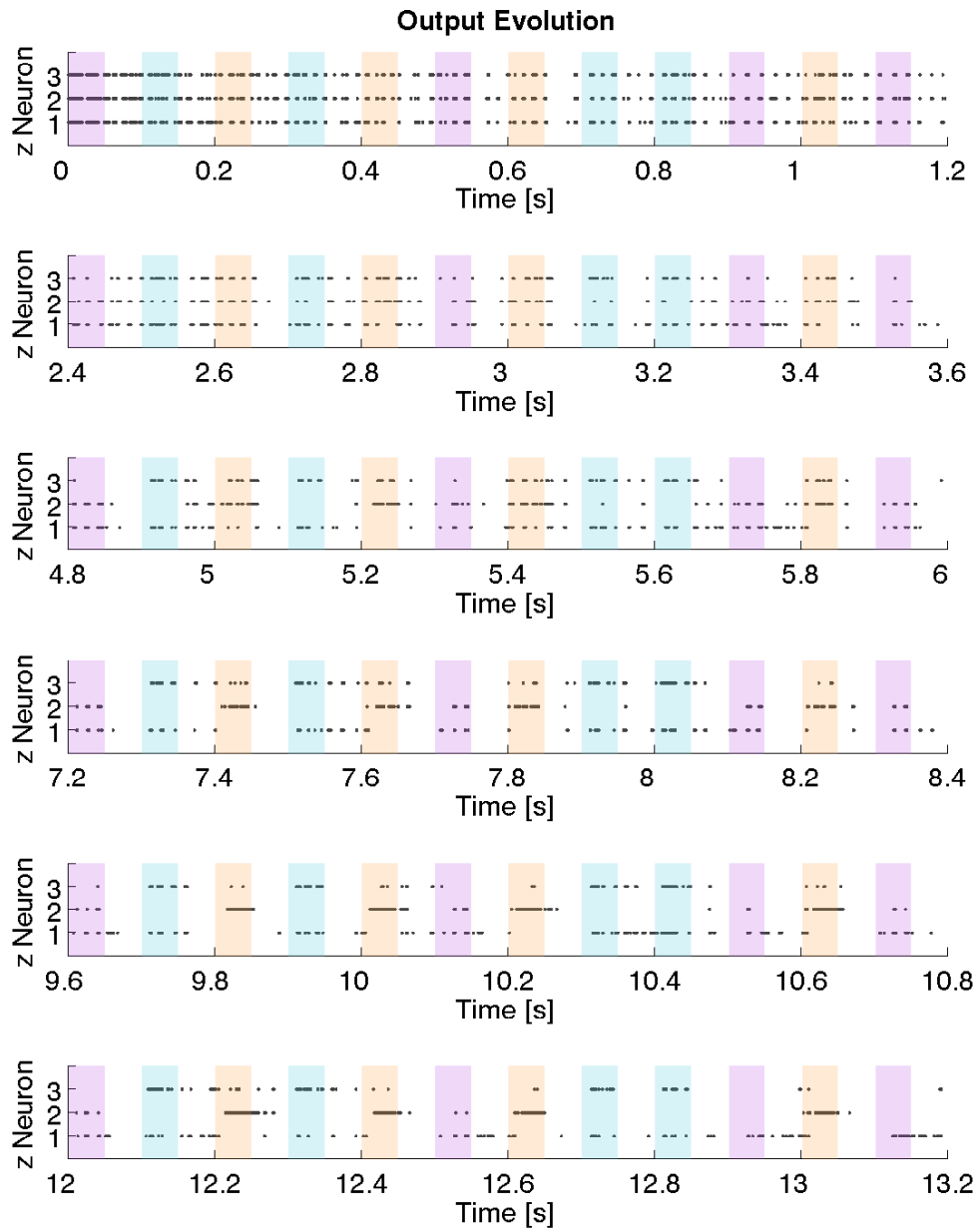


Figure 4.3: Output spike evolution while training a network with $K = 3$ output neurons and $L = 3$ different patterns. Continued in Figure 4.4.

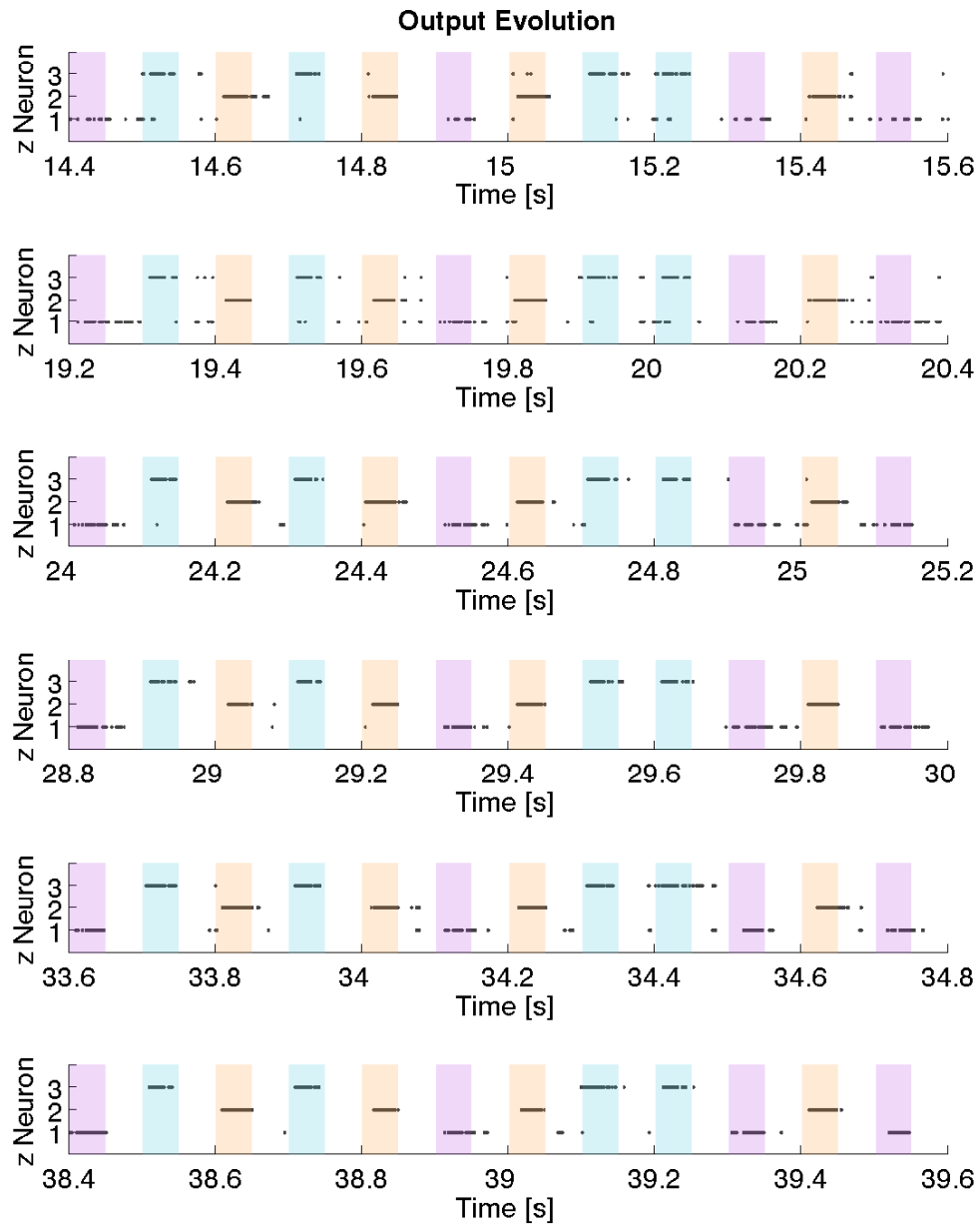


Figure 4.4: Continued from Figure 4.3. Output spike evolution while training a network with $K = 3$ output neurons and $L = 3$ different patterns.

input weights are initialized randomly, all **z-Neurons** start to fire approximately equally. After a while, the network starts to detect the repeated patterns and spike trains begin to be limited to these time frames.

At the same time, the competition between the output neurons already suppresses other **z-Neurons** when one is already a bit stronger for a specific pattern. This sharpening of the classification contrasts takes until the end of Figure 4.4 where the classification is already working properly.

4.6 Classification Performance

4.6.1 Uniform Pattern Distribution with a **z-Neuron** Surplus

This configuration could be seen as the most basic and reasonable one. With a number of **z-Neurons** to a number of patterns ratio of

$$\# \text{ of } \mathbf{z}\text{-Neurons } K \geq \# \text{ of Different Patterns } L, \quad (4.6)$$

all presented input patterns could basically be separated, as enough **z-Neurons** are available. For the basic classification analysis, patterns are selected with equal probability for each one. Therefore, each available pattern has been presented during training for approximately the same time.

With simulations being done with respect to the given properties, classification performances of $\approx 100\%$ can be reached.

A comparison between network output activities of a theoretical **SEM** model and a biological designed model can be seen in Figure 4.5. The simulated network configuration was with $K = 4$ **z-Neurons** and $L = 3$ different input patterns.

Output of the reference **SEM** model can be seen in Figure 4.5(a). Each of the **z-Neurons** specializes on a different pattern, with noise counting as an additional pattern. Therefore, the number of patterns is equal to the number of **z-Neurons**. The model manages to keep the output rate approximately constant over all **z-Neurons**. Once a pattern is applied to the input, it takes some onset time for the network to detect the change and for the specific neuron to react. During this time, the previously active neuron stays active and keeps spiking. Therefore, the spike train indicating a pattern is delayed from the on and off times from the pattern.

Figure 4.5(b) shows the actual biologically inspired spiking **WTA** model. At first three major differences to the **SEM** model come to sight:

1. Immediately when starting the simulation, every output neuron starts bursting spikes. This is due to the spiking inhibition loop, where the inhibitory neuron first needs excitatory input before it actually starts emitting spikes itself. But once the inhibition becomes effective, only the patterns representative neuron keeps spiking. The bursting behavior of an active output neuron can be due to inhibition gain and membrane potential offsets. Once an input pattern is finished, the specific neuron stops firing nearly immediately (depending on how good the pattern's final part can be distinguished from noise).
2. Once noise is presented to the network, all of the **z-Neurons** stop firing completely. As random noise can not raise the **z-Neurons**' membrane potentials high enough to get them spiking, the network stays silent. This inherent noise filter may provide welcome functionality, but is problematic when wanting to introduce new patterns once the network is already trained. This would have to be taken care of by adjusting resting potentials and injected noise to make the spiking more probabilistic.
3. As the noise is filtered out by the the biological model, there is a surplus over the SEM model of one of the **z-Neurons** for pattern classification. Therefore, the neuron starts to split a pattern apart and classifies the first, as well as the second half of, in this example, pattern #2 as separate patterns. This is the same behavior that a SEM model would exhibit when an additional output neuron would be introduced.

For a better understanding of the differentiation between the patterns done by the **z-Neurons**, Figure 4.6 shows their membrane potentials. While the membrane potential v_k of an active neuron remains above 0, the others are suppressed by the inhibition. This has the effect that just one neuron is spiking as intended for WTA behavior.

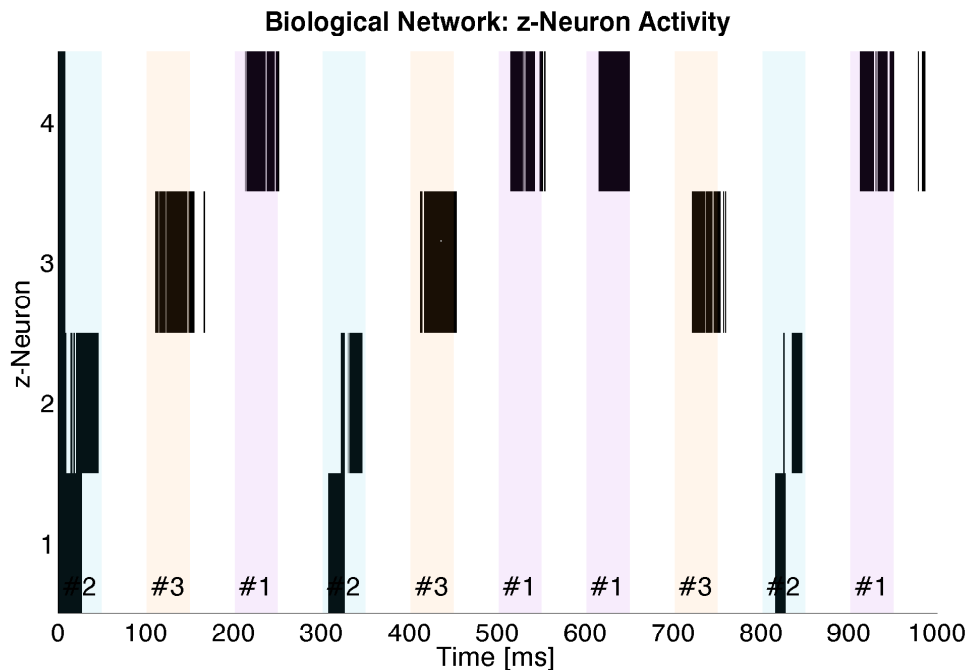
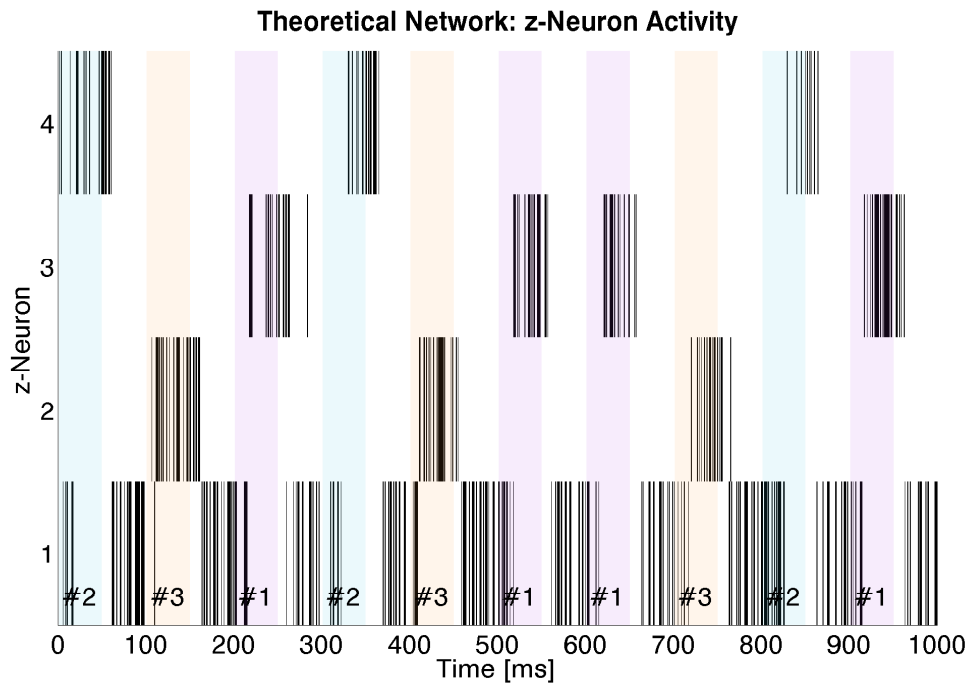


Figure 4.5: z-Neuron surplus: Output activity of a network with $K = 4$ z-Neurons and $L = 3$ different input patterns. (a) For a theoretical model. (b) For a biological model.

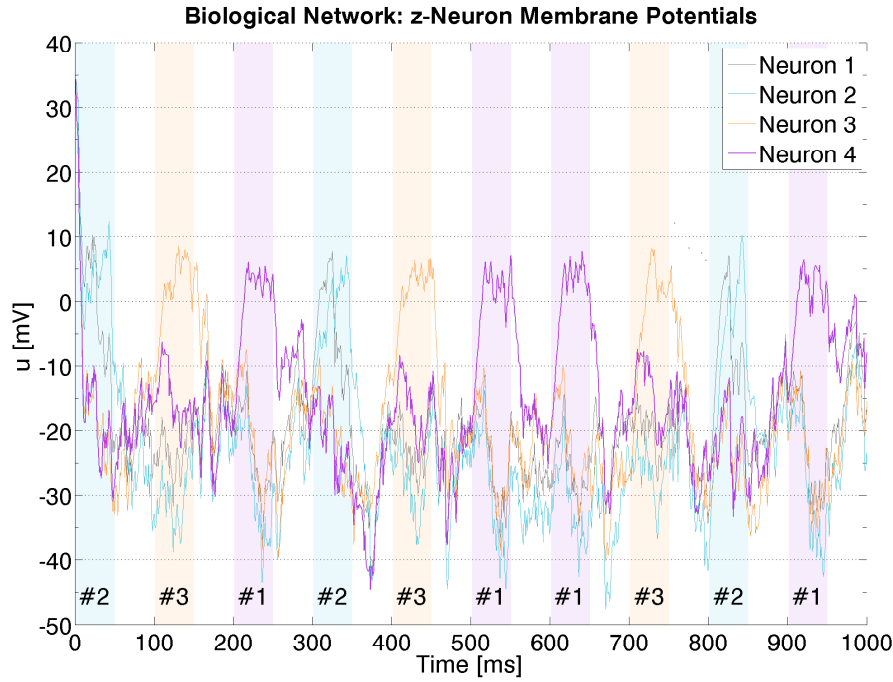


Figure 4.6: Membrane potentials v_k for the same biological network as in Figure 4.5(b).

4.6.2 Uniform Pattern Distribution with a z-Neuron Shortage

Limiting the available z-Neurons is a second test case scenario. With a number of z-Neurons to a number of patterns ratio of

$$\# \text{ of z-Neurons } K < \# \text{ of Different Patterns } L \quad (4.7)$$

not every available pattern can be represented by a unique output neuron. This case may not be of equal importance for specifically designed networks that should suit a predefined pattern set with the aim of classifying available patterns the best possible. But while using pattern sets of unknown size, the number of z-Neurons can only be estimated. Furthermore, additional patterns may be introduced while the network is already working (which would be the biologically most interesting way). The ability of the network to classify, and therefore its classification performance, depends highly on how short the network is of z-Neurons in comparison to the number of patterns L .

An example comparison between network output activities of a theoretical SEM model and a designed biologically plausible model can be seen in Figure 4.7. The simulated

network had $K = 3$ z-Neurons and $L = 7$ different input patterns.

Again, the output activity of a trained SEM model serves as a reference. The activity of its z-Neurons can be seen in Figure 4.7(a). The performed classification is very simple to analyze. As long as there are z-Neurons available (not responsible for a pattern yet), one of the free ones specializes on the pattern first presented when starting to train the network (learn its input weights). In this simulation this happens for z-Neurons 1 and 2 specializing on patterns #1 and #3 as these had been the first randomly selected when the training simulation was started. To keep the overall rate constant, the remaining neuron becomes active for every other pattern (including noise). No further discrimination between patterns can happen as there are no z-Neurons left. The neuron picks patterns as long as z-Neurons are available, with the last of the z-Neurons treating all remaining patterns equal to noise.

The simulation results of z-Neurons' activity of the biological network can be seen in Figure 4.7(b). The classification is related to the previous one of the SEM model in terms of treating the early presented patterns in training equal. Again, as z-Neurons are available, recently introduced patterns can be assigned to them. As patterns #1 and #3 are presented first in training, they get responsive neurons that mark their occurrence. Again the biological spiking network shows two model-inherent differences:

1. As already described in Section 4.6.1, noise is not treated as an additional pattern, but left out completely of classification (inherent noise filtering). With too few z-Neurons to classify all different patterns, excessive patterns get treated as noise and therefore, the network remains silent while one of these patterns is active. This leaves one additional z-Neuron available for representing a single pattern (in comparison to the SEM model).
2. The activities of z-Neurons do not necessarily exclude each other. Thus, it is possible for more than one of the z-Neurons to specialize on the same input pattern, as it happened in the plotted example for pattern #1. When initial input weights to different z-Neurons are close enough, both (or more) can start firing simultaneously and therefore learn to trigger on the same input pattern. As z-Neurons do not have exact feedback information as in the SEM model, this behavior is inherent and can occur for different network configurations (not only for a z-Neurons shortage). The neurons might fire with a lower rate than when mutually excluded, but both remain active though inhibition.

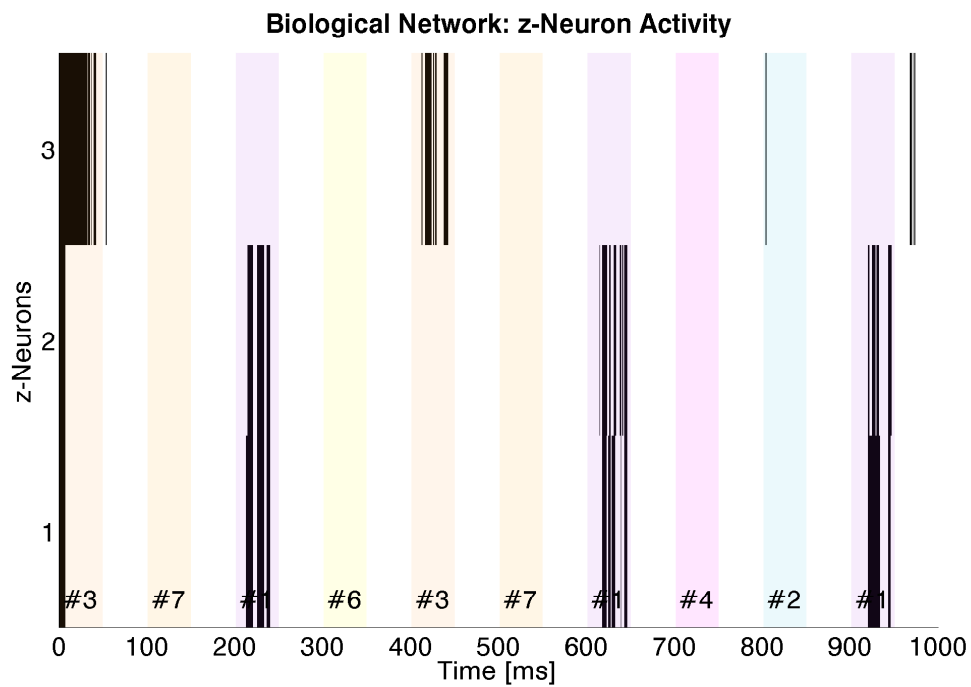
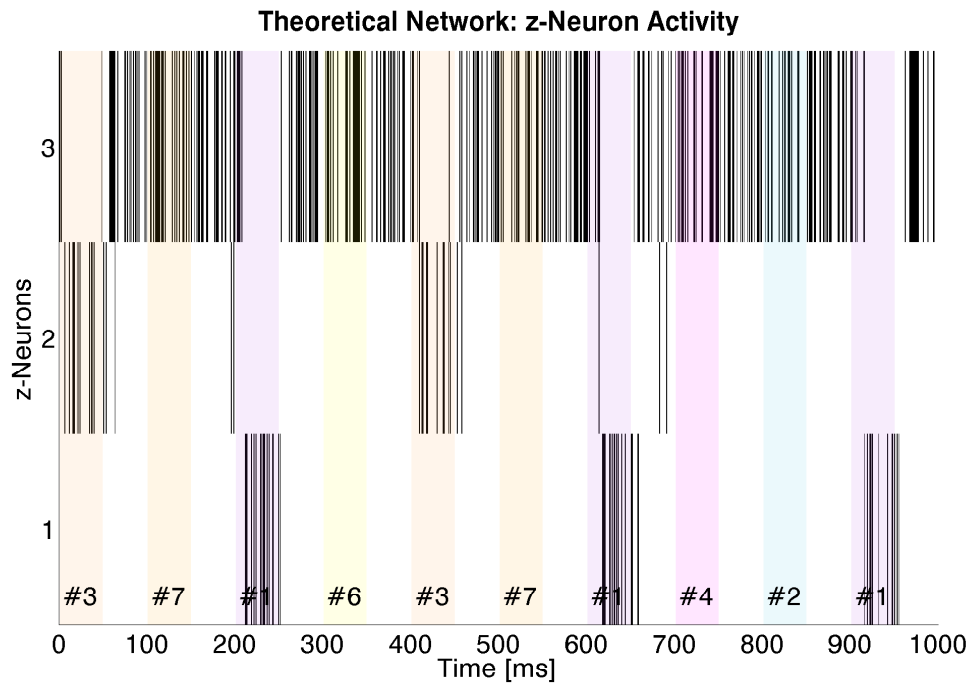


Figure 4.7: z-Neuron shortage: Output activity of a network with $K = 3$ z-Neurons and $L = 7$ different input patterns. (a) For a theoretical model. (b) For a biological model.

4.6.3 Uniform Pattern Distribution: Number of z-Neurons vs. Number of Patterns

Figure 4.10 summarizes the network's dependence on the ratio of the number of z-Neurons K to the number of patterns L in comparison to SEM output in Figure 4.9. Contrary to the SEM, model (Figure 4.9(a)), the designed WTA circuit (Figure 4.10(a)) can still separate a single input pattern from all others with just one z-neuron. As the rate is kept constant over time performing SEM no differentiation between any patterns can be made. Due to its implicit noise cancellation, an additional neuron for classification is available in the biological model.

Both networks perform equally well (Figure 4.9(b) and Figure 4.10(b)) in terms of classification, with the number of z-Neurons equal to the number of patterns. Again, noise is left out in classifications from the proposed model. This helps to determine the time a pattern really is active on the input.

With a neuron surplus both networks (Figure 4.9(c) and Figure 4.10(c)) start to split the patterns into respective sub-patterns.

4.6.4 Non-Uniform Pattern Distribution

Previous results showed that the designed WTA network can distinguish patterns with an uniform occurrence probability (all patterns occur equally likely) very well. Additionally, a prior-probability is introduced to make the occurrence of a pattern more or less likely. Since the output rate of the biological network is not kept constant over time (as the network remains silent for noise), unlikely input patterns can turn out to be a problem. As they are rarely presented, the network treats them as noise and their occurrence gets lost. Input weights are then adapted to suit the most likely patterns.

The dependence of the classification performance on the occurrence probability is demonstrated in Figure 4.8. Simulations of a network with $K = 3$ z-Neurons and $L = 2$ different input patterns were performed and their classification performance evaluated over 10 trials each. The occurrence probability of pattern #1 $p(\#1)$ thereby varied from 10% to 50%. Pattern #2 had therefore an occurrence probability of

$$p(\#2) = 1 - p(\#1). \quad (4.8)$$

Starting with 10% occurrence probability of pattern #1, the biological network's classification performance is at approximately 65%, which is hardly better than having learned just

a single pattern. With the occurrence probabilities becoming more equal, the classification performance increases and reaches $\approx 100\%$ for uniformly presented input patterns.

This dependence on the distribution of input patterns is a profound problem of the designed network, that would have to be addressed by introducing adapting inhibition.

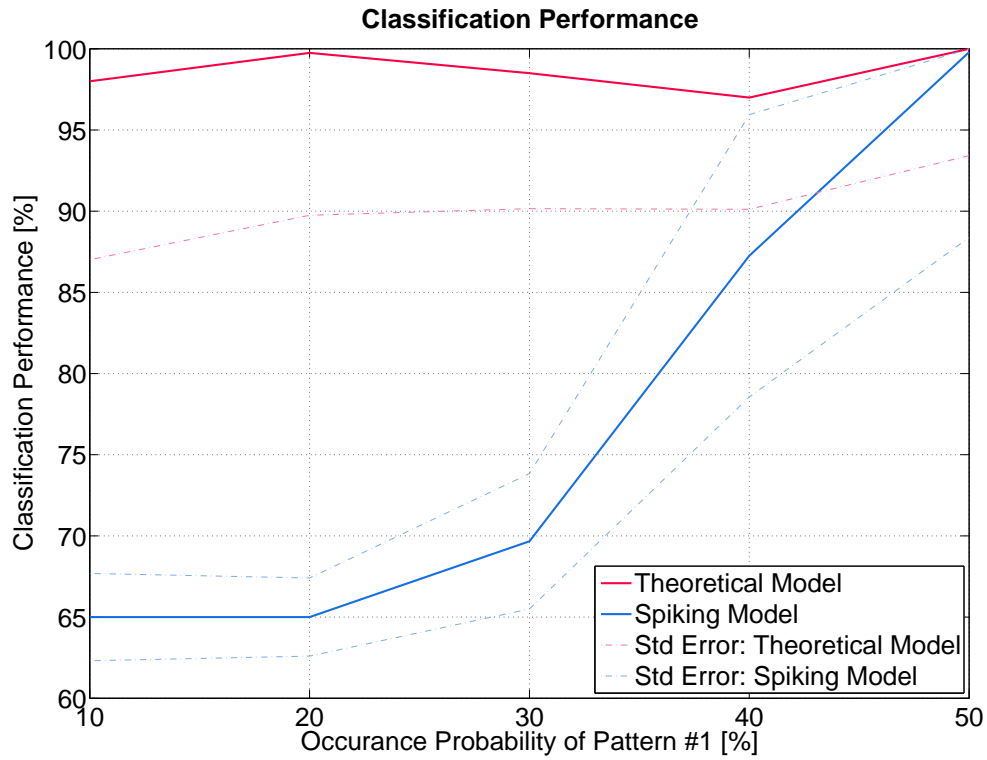
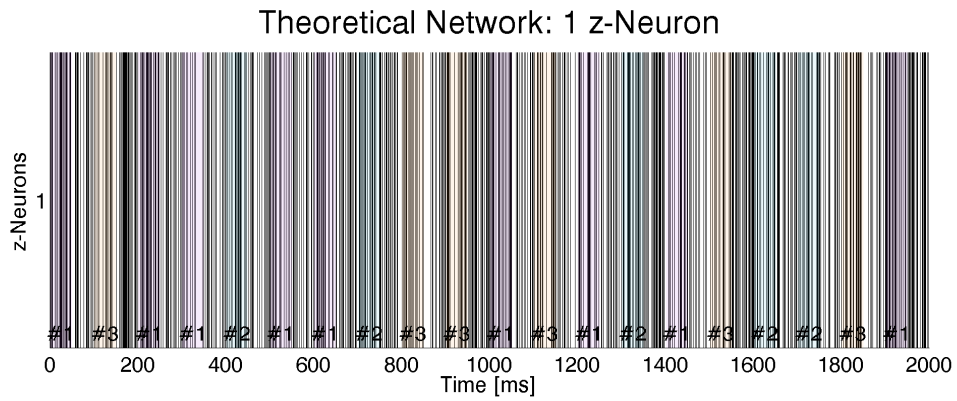
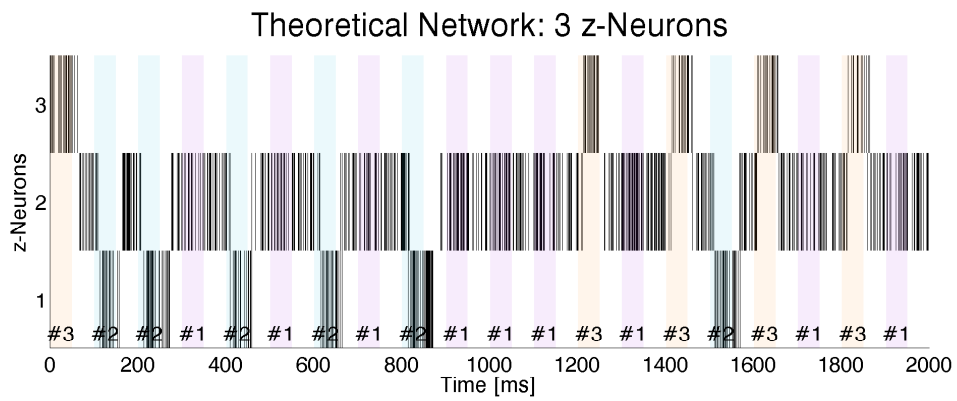


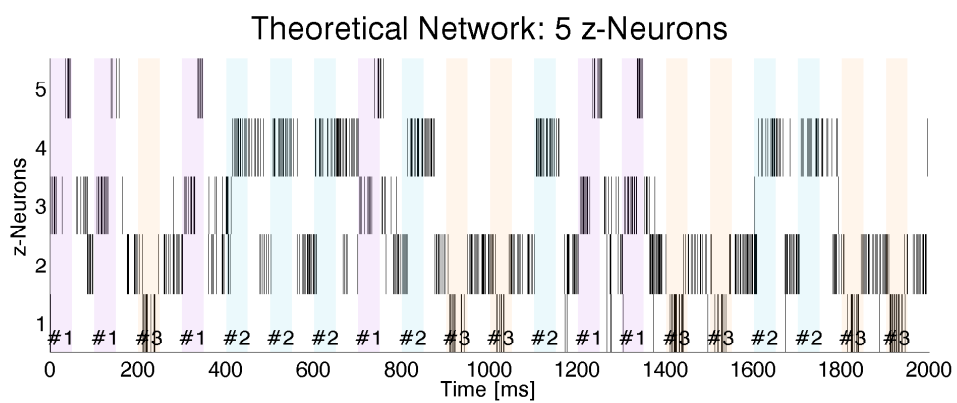
Figure 4.8: Classification performance of a network in dependence of occurrence probability of pattern #1. Each setup was simulated over 10 trials, with $K = 3$ output neurons and $L = 2$ input patterns.



(a)

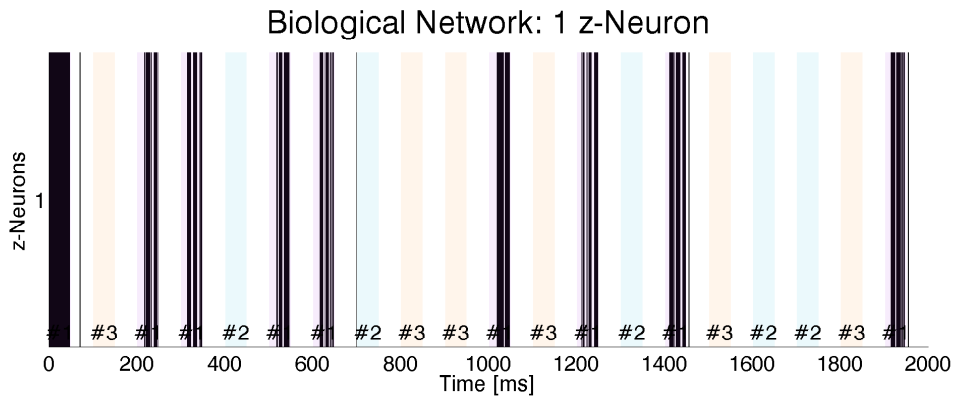


(b)

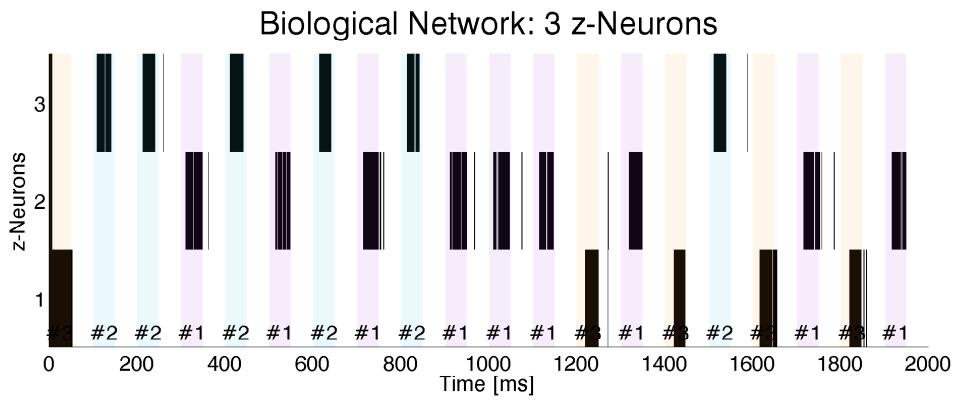


(c)

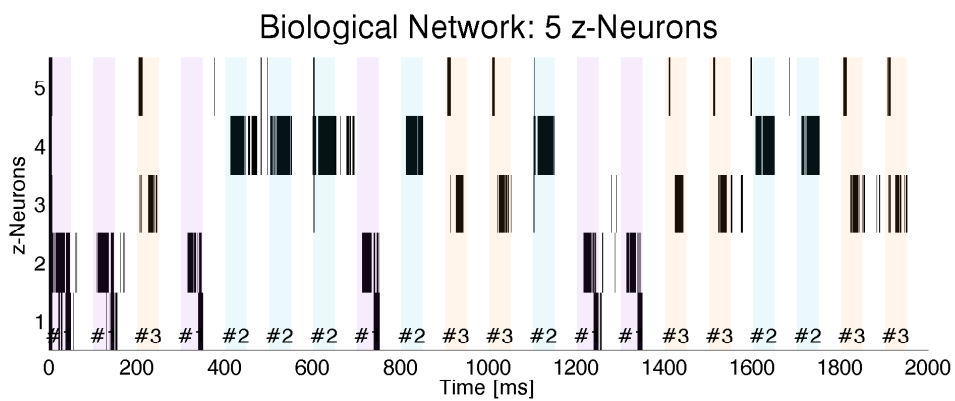
Figure 4.9: Theoretical Network: 3 different patterns get clustered by (a) 1 z-neuron, (b) 3 z-neurons and (c) 5 z-neurons.



(a)



(b)



(c)

Figure 4.10: Biological Network: 3 different patterns get clustered by (a) 1 z-neuron, (b) 3 z-neurons and (c) 5 z-neurons.

4.7 Generative Model Validation

4.7.1 Uniform Pattern Distribution without Noise Breaks

The basic setup to analyze a generative model can be constructed with equally likely (uniformly distributed) input patterns without any noise breaks. For simulation, a network with $K = 3$ z-Neurons and $L = 3$ different input patterns was used.

As already mentioned, the learning itself has to be finished to evaluate the model. Figure 4.11 shows how the input weights of both, a theoretical SEM model and a biological model, evolve. After some time, learning has finished and the weights have converged. The developed model can be analyzed.

As we can not give an absolute performance measure of how good the learned model is, the model learned by a SEM is set as reference. Figure 4.12(a) shows the $\ell\ell$ of both models. Additionally, the $\ell\ell$ of the same models for randomly permuted inputs is printed to give a lower-performance bound. The generative model of the biological network is nearly as good an approximation of the original model as the SEM generated one. Further, both $\ell\ell$ values are far above the randomly permuted lower-bound. Therefore, the model has learned properly.

Figure 4.12(b) shows the fluctuation of the output rates. Both network's output rates fluctuate, but evolve to keep a stable mean value. The trend of the output rates does not change after the initial learning phase.

4.7.2 Uniform Pattern Distribution with Noise Breaks

Additionally to the setup of Section 4.7.1, noise breaks of equal length as the patterns are separating individual pattern occurrences. Again, the $\ell\ell$ does only provide a meaningful measure once the input weights have converged and the generative model remains unchanged (Figure 4.13).

The designed networks $\ell\ell$ again grows towards the SEM reference, but both values are closer to the lower-bounds than in noise free simulations. The model learned can still be seen as a valid input model approximation.

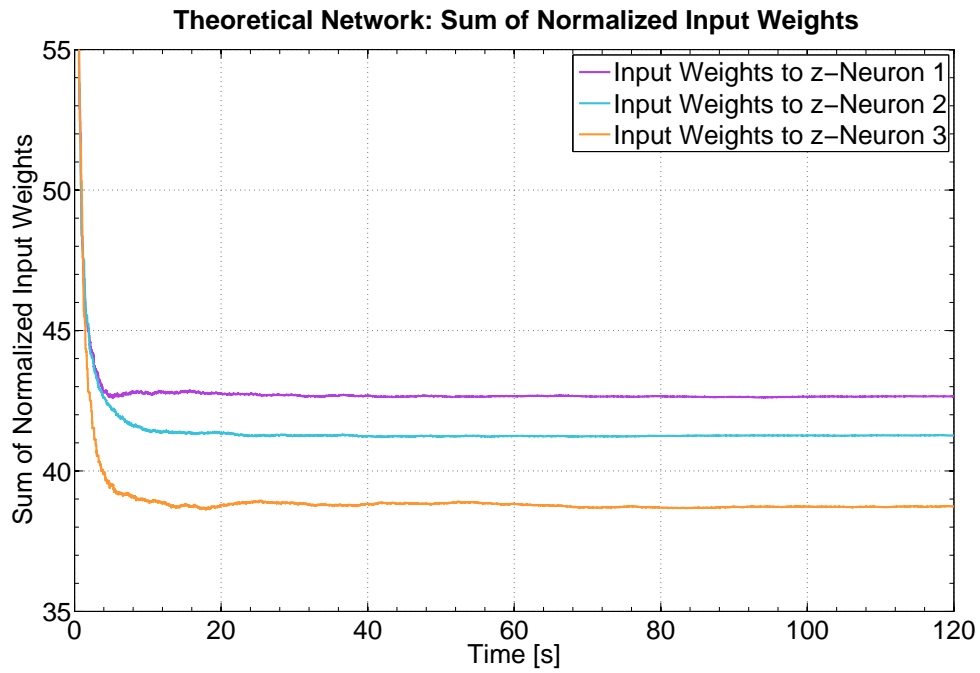
4.7.3 Non-Uniform Pattern Distribution with Noise Breaks

For the next simulations, the input patterns are selected with unequal probabilities

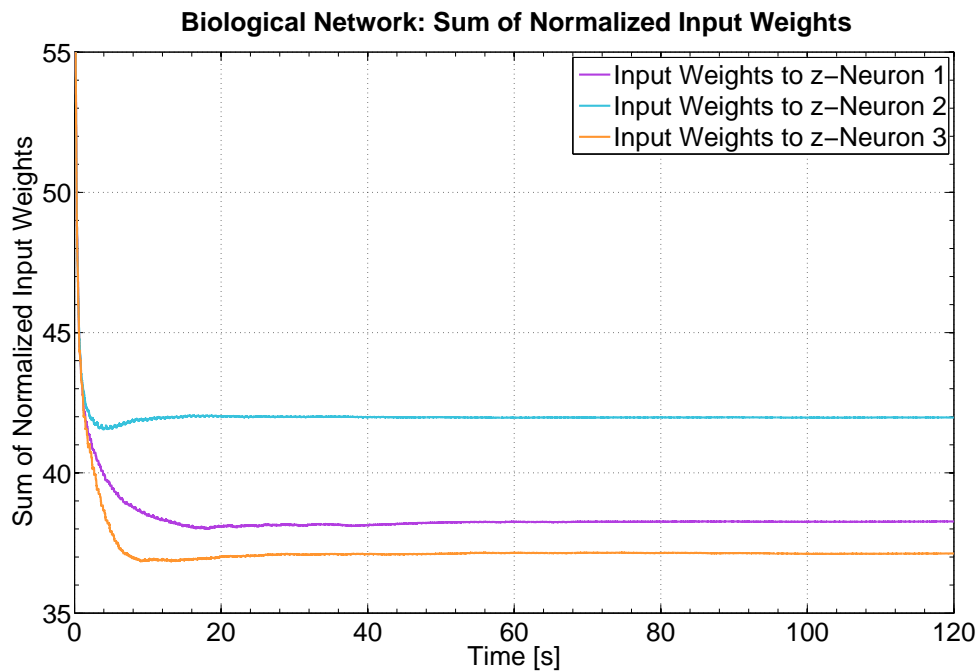
$$p(\#1) = \frac{6}{12} \quad p(\#2) = \frac{4}{12} \quad p(\#3) = \frac{2}{12}. \quad (4.9)$$

This unequal presentation of the different patterns already caused some difficulties in Section 4.6.4 when the network's output was observed. However, a direct inference can not be made. Once again, only the final part of the plots can be evaluated as learning has to be finished.

Figure 4.14(b) shows the $\ell\ell$ of the network trained with the unequally presented input patterns. The $\ell\ell$ of the biological model drifts far apart from the theoretical SEM models $\ell\ell$. It is even below the lower-bound of the SEM model. This indicates that the model learned is very different from the original input distribution, as the $\ell\ell$ is even below a trained SEM model with randomly connected inputs. As mentioned, this does not necessarily mean that some patterns can be separated, but the generative model learned is by far not as good as the SEM ones.

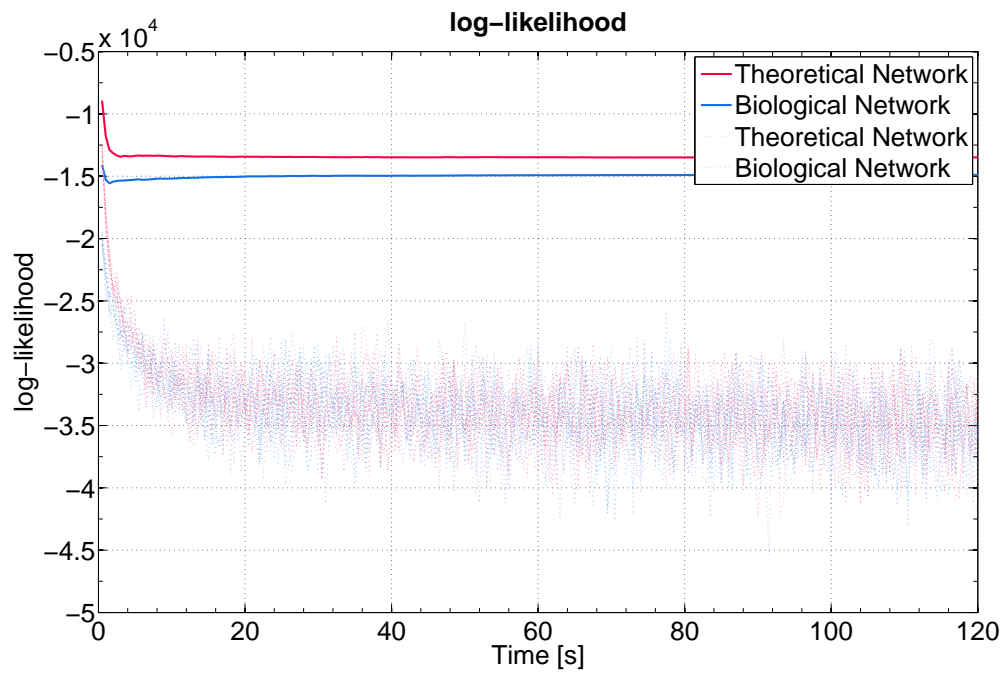


(a)

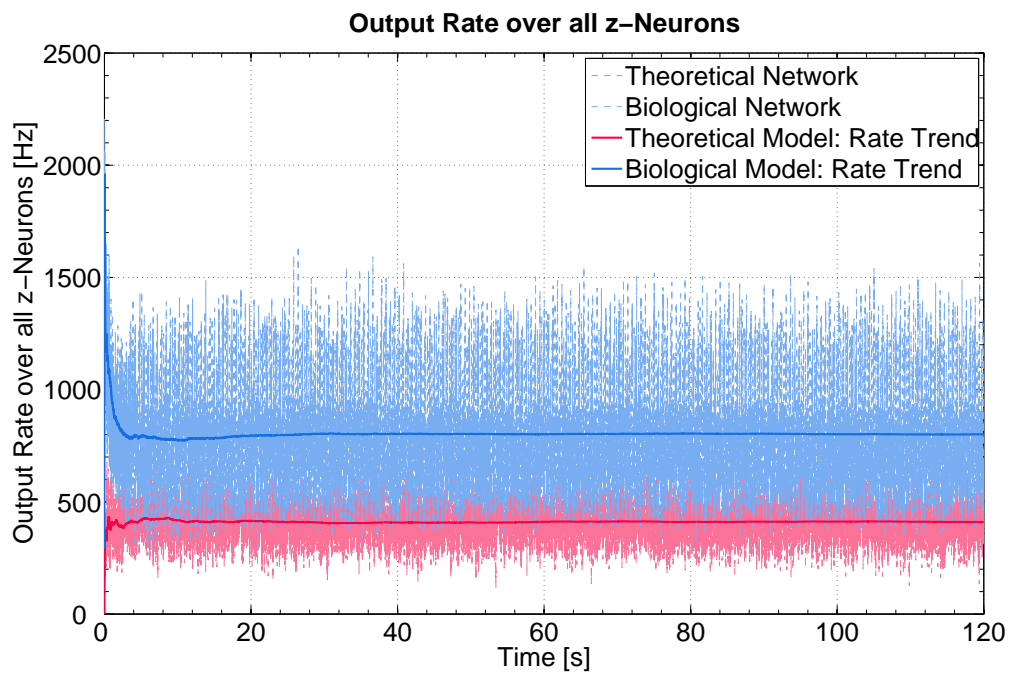


(b)

Figure 4.11: Sum of normalized input weights while learning uniformly distributed input patterns without noise breaks.

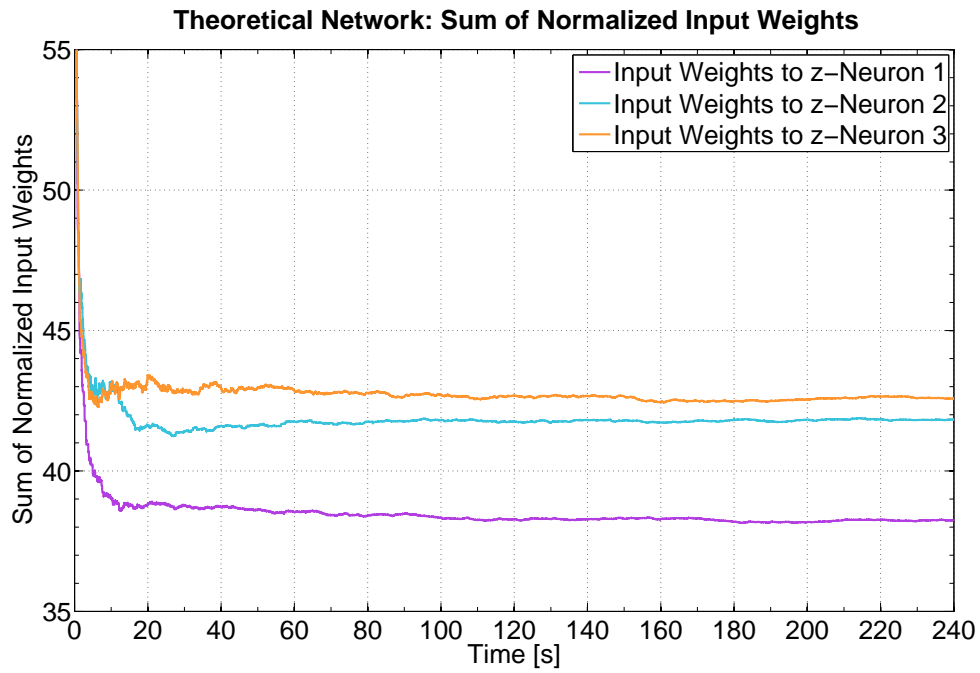


(a)

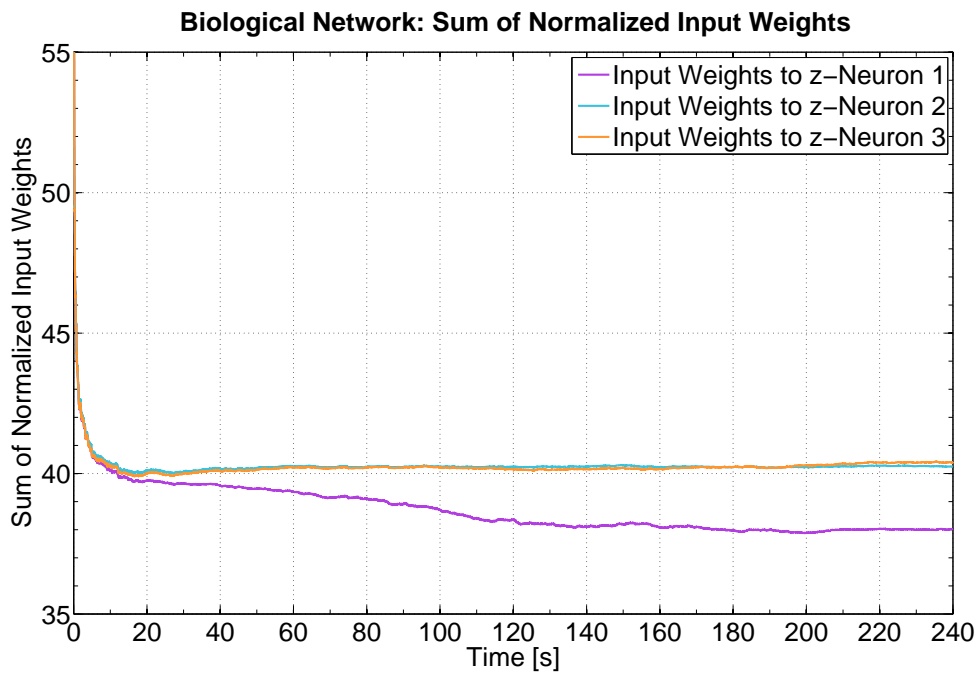


(b)

Figure 4.12: Uniform distributed input patterns without noise breaks: (a) Log-Likelihood during training. (b) Output rates of z-neurons during training.

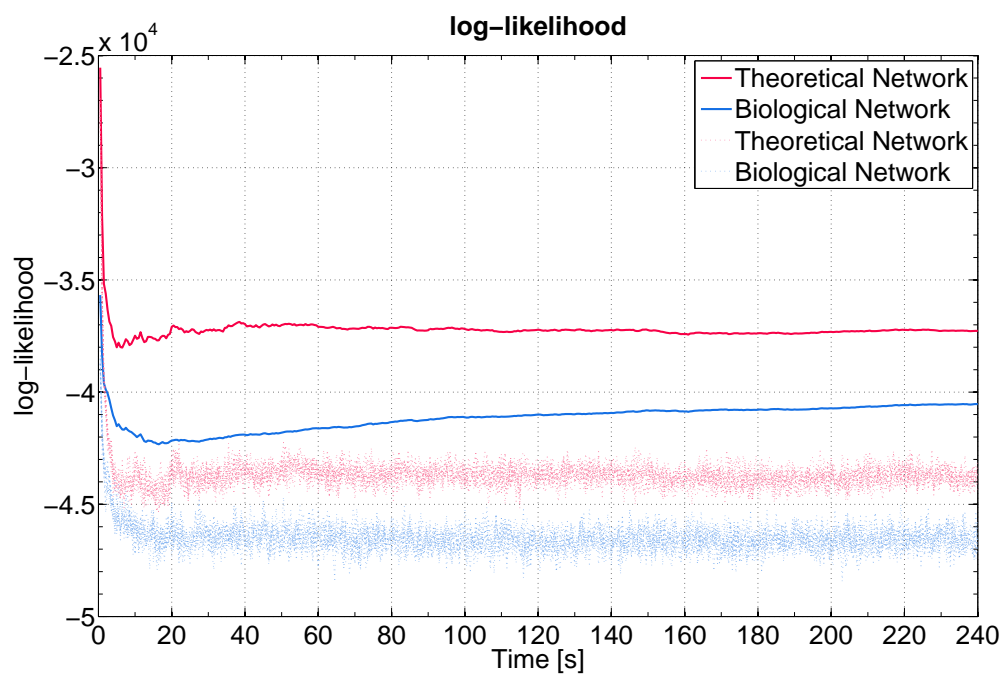


(a)

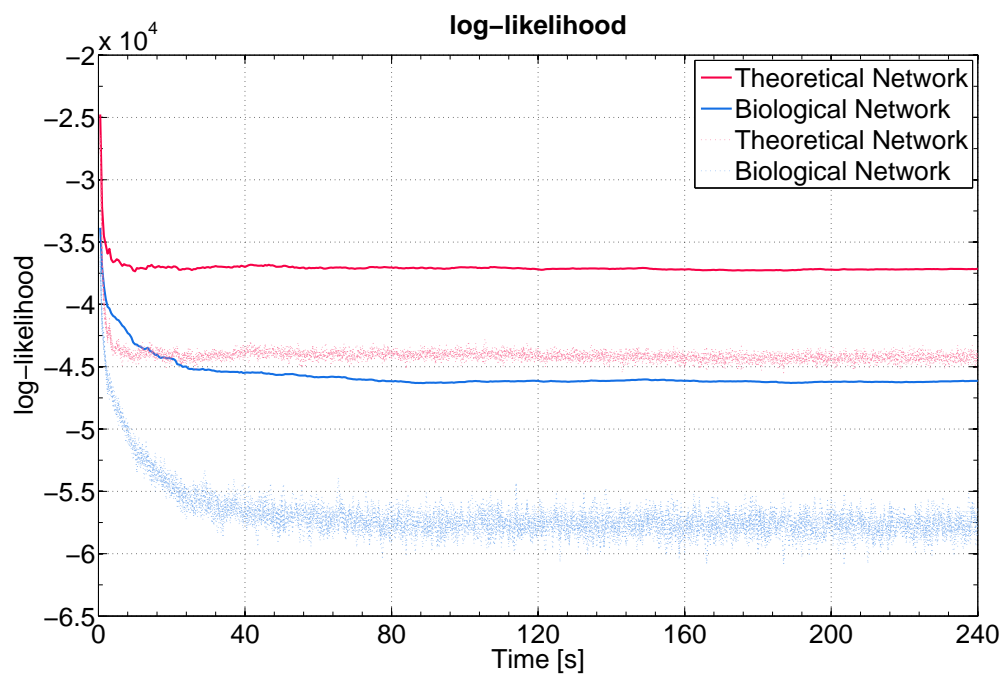


(b)

Figure 4.13: Sum of normalized input weights while learning uniformly distributed input patterns with noise breaks separating pattern occurrences.



(a)



(b)

Figure 4.14: Log-Likelihood of network while training patterns with noise breaks with (a) uniform input distribution, (b) non-uniform input distribution.

4.8 Exponential Inhibitory Neuron

As further step into the direction of a biologically realistic model, the WTA network's linear inhibitory neuron could be replaced by an exponentially firing one [18]. Classification is still working while using an exponential neuron, but the previous analysis was done using a linear one for the sake of simplicity and its closer relation to the SEM theory.

Figure 4.15 shows an example plot of a network trained and tested with z-Neurons with an output rate exponentially depending on their membrane potentials. As desired, a trained network is again able to differentiate between individual patterns. This demonstrates the basic ability of the WTA network to learn, even with an exponential inhibitory neuron.

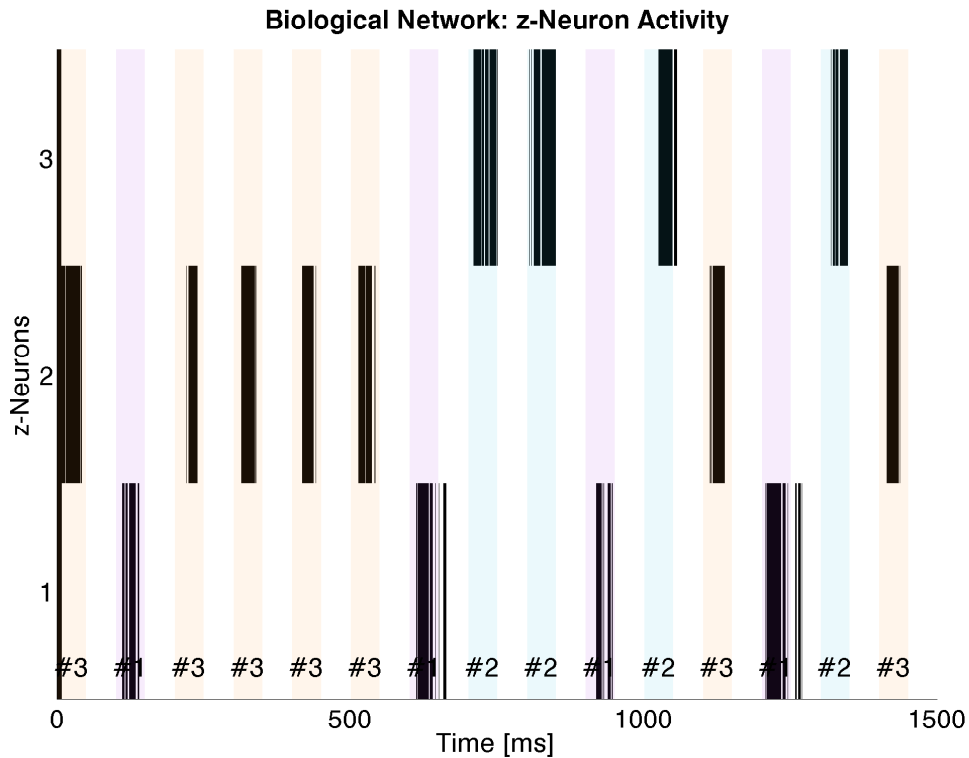


Figure 4.15: Classification with an exponential inhibitory neuron.

Chapter 5

Conclusion and Future Work

Contents

5.1 Conclusion	53
5.2 Future work	55

5.1 Conclusion

In this master’s thesis a WTA circuit using realistic neuron models for the discovery of hidden causes (via STDP) was presented. At first the basic concepts of WTA and STDP were recapitulated.

Chapter 2 gave a very brief introduction to the theoretical work of Nessler, Pfeiffer and Maass [27, 28] that motivated this thesis. The desired form of soft WTA competition (Section 2.1.1) that should be remodeled using a completely spiking feedback loop instead of theoretical normalization was shown. Further, it was argued in Section 2.2 that the neuron models used for computer simulations within this thesis are a valid phenomenological approximation of biologically realistic neurons.

The designed WTA circuit was introduced in Chapter 3 that uses a known architecture [29]. The design of individual components within the proposed network was done to create a good approximation of the theoretical model [27, 28], while keeping models of neurons and synaptic transmission within a biologically reasonable range. However, it was not easy at all to establish stable output behavior in terms of weight convergence. Crucial parameters had to be identified, parameter ranges defined and their values fine-tuned with the use of computer simulations.

The results of extensive computer simulations were presented in Chapter 4. It became obvious that the proposed WTA circuit can approximate SEM behavior within limitations:

- For uniformly distributed input patterns, the proposed network approximates the SEM model very well (Section 4.7.1). The log-likelihood of the generative model learned is close to the SEM. The network can distinguish individual trained input patterns with a classification performance of $\approx 100\%$ (for network sizes as given in Chapter 4).
- For patterns that are not equally likely presented while training, the performance is limited. Patterns that are presented far less likely can get treated as uncorrelated noise (when introduced at a later stage of training) and are therefore completely left out of classification (this peculiarity of the designed WTA circuit is described below in more detail). The log-likelihood in this case is below the one of a randomly connected SEM model. In this case, the generative model has only learned those patterns that were presented more often what is a clear disadvantage in comparison to the theoretical model. Although the generative model learned does not fit the input, the network can still be able to perform classification.

The major difference between the theoretical model and the designed WTA circuit is the z-Neurons' membrane potentials of which the inhibition is computed. Theoretical inhibition can be computed according to Equation (2.2) and applied immediately. Due to biological constraints, basic differences are:

- The z-Neurons' membrane potentials u_k created only through inputs can not be accessed directly by the inhibitory neuron.
- The inhibitory neuron can only sample a value of the z-Neurons' membrane potentials through spikes they emit.
- As the inhibitory feedback to the z-Neurons affects their membrane potentials, they fire dependent on their complete membrane potential v_k and not only the part of it generated through input u_k . Therefore, the inhibition of the spiking model has to drift apart from the theoretical one that is computed with direct access to u_k .
- Theoretical inhibition is applied immediately, but through spike transmission times, delays are introduced in a spiking feedback loop. Due to this, the applied inhibition always lacks a bit behind.

The designed WTA circuit shows a few differences in the spiking output behavior of its z-Neurons compared to a theoretical SEM implementation:

- The proposed network shows an inherent noise filter. Once the input weights are adapted to specific patterns, z-Neurons remain silent when noise is presented at the input.

On the positive side, this helps to get sharper contrasts to indicate when a pattern is on or off. As no neuron is occupied with being a specialist for classifying noise, an additional neuron remains to distinguish patterns. Therefore, the proposed network can theoretically cluster a pattern more than a SEM model of the same size, as noise is indicated with silence. This is especially useful when the number of z-Neurons is less than or equal to the number of different patterns. In the extreme case of just a single output neuron, the designed WTA circuit is still able to distinguish a single pattern from the rest.

On the other hand, this behavior can cause the biggest disadvantage of the proposed model. As some early presented patterns are trained, less likely ones get treated like noise when presented later in the training phase, and are therefore completely left out of classification. Problems with non-uniform distributions of input patterns arise.

- As there is no mutual exclusion of two or more z-Neurons firing at a time, it can happen that likely initialized input weights to more than one of the z-Neurons can cause them to start firing almost at the same time. Consequently, a single pattern can be learned from more than one output neuron.

We can conclude that the WTA circuit proposed in this thesis provides a well-suited, completely spiking approximation of the SEM theory for a defined set of uniformly distributed input patterns. In this case, the generative model learned is as good as the one learned by SEM. This approximation does not work for alterations in the input distribution. Although the network can still perform classification, an invalid generative model is learned.

5.2 Future work

It would be desired to investigate how introducing an adaptation mechanism in the inhibitory loop (neuronal or synaptical), would help to deal with non-uniform input dis-

tributions. Additional robustness to changes in the input distribution would be a future goal. Or even further robustness to changes in the internal structure of the WTA circuit and its ability to repair itself.

A highly desirable feature would be to port the network onto spiking neuromorphic hardware. The hardware developed during the FACETS* project would be a possible target.

The designed circuit shall be extended in terms of biological plausibility in computer simulations with more detailed models for excitatory and inhibitory neurons and the modelling of receptors for specific synapse types.

*<http://facets.kip.uni-heidelberg.de/>

Appendix A

Acronyms and Symbols

List of Acronyms

AP	Action Potential
ARP	Absolute Refractory Period
EM	Expectation Maximization
EPSP	Excitatory PostSynaptic Potential
IPSP	Inhibitory PostSynaptic Potential
LPF	Low-Pass Filtered
LTD	Long-Term Depression
LTP	Long-Term Potentiation
PSP	PostSynaptic Potential
RRP	Relative Refractory Period
SEM	Spike-based Expectation Maximization
SRM	Spike Response Model
STDP	Spike-Timing Dependent Plasticity
WTA	Winner-Take-All
z-Neurons	Output Neurons

List of Symbols

Δ^{ax}	Axonal spike transmission delay.
$\epsilon(s)$	PSP kernel.
$\mathcal{I}(t)$	Inhibition.

$\ell\ell$	log Likelihood.
$\lambda_k(t)$	Theoretical output rate of neuron k without inhibition.
$\Lambda_k(t)$	Effective output rate of neuron k .
τ	Time constant for the exponential decay of a PSP.
$\Theta(s)$	Heaviside step function.
$f_k(t)$	Poisson rate of neuron k .
u_0	Membrane resting potential.
$u_k(t)$	Theoretical membrane potential of neuron k without inhibition.
$v_k(t)$	Effective membrane potential of neuron k .
$w_{k,0}$	($\hat{=}u_0$) Membrane resting potential of neuron k .
x_{Inh}	Inhibitory neuron.
y_i	Input neuron i .
z_k	Output (z) neuron k .

Bibliography

- [1] Aviel, Y. and Gerstner, W. (2006). From spiking neurons to rate models: A cascade model as an approximation to spiking neuron models with refractoriness. *Physical Review E*, 73(5):1–10.
- [2] Bi, G. and Poo, M. (2001). Synaptic modification by correlated activity: Hebb’s postulate revisited. *Annual Review of Neuroscience*, 24:139–66.
- [3] Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Springer, New York.
- [4] Brette, R. and Gerstner, W. (2005). Adaptive exponential integrate-and-fire model as an effective description of neuronal activity. *Journal of Neurophysiology*, 94(5):3637–42.
- [5] Caporale, N. and Dan, Y. (2008). Spike timing-dependent plasticity: a Hebbian learning rule. *Annual Review of Neuroscience*, 31(February):25–46.
- [6] Clopath, C., Jolivet, R., Rauch, A., Luescher, H., and Gerstner, W. (2007). Predicting neuronal activity with simple models of the threshold type: Adaptive Exponential Integrate-and-Fire model with two compartments. *Neurocomputing*, 70(10-12):1668–1673.
- [7] Dan, Y. and Poo, M.-M. (2004). Spike timing-dependent plasticity of neural circuits. *Neuron*, 44(1):23–30.
- [8] Debanne, D. (2004). Information processing in the axon. *Nature reviews. Neuroscience*, 5(4):304–16.
- [9] Dempster, A., Laird, N., and Rubin, D. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):1–38.
- [10] Douglas, R. J. and Martin, K. A. C. (2004). Neuronal circuits of the neocortex. *Annual Review of Neuroscience*, 27:419–51.
- [11] Gerstner, W. (1999). Spiking Neurons. In Maass, W. and Bishop, C. M., editors, *Pulsed Neural Networks*, chapter 1, pages 3–54. The MIT Press.
- [12] Gerstner, W. and Kistler, W. M. (2002). *Spiking Neuron Models*. Cambridge University Press.

- [13] Gerstner, W. and Naud, R. (2009). How Good Are Neuron Models? *Science*, 329:379–80.
- [14] Hahnloser, R., Sarpeshkar, R., Mahowald, M., Douglas, R., and Seung, H. (2000). Digital selection and analogue amplification coexist in a cortex-inspired silicon circuit. *Nature*, 405(6789):947–951.
- [15] Hebb, D. O. (1949). *The Organization of Behavior*. Wiley, New York.
- [16] Hodgkin, A. and Huxley, A. (1952). A quantitative description of membrane current and its application to conduction and excitation in nerve. *The Journal of Physiology*, 117:500–544.
- [17] Jolivet, R., Rauch, A., Lüscher, H., and Gerstner, W. (2006a). Integrate-and-Fire models with adaptation are good enough: predicting spike times under random current injection. *Advances in Neural Information Processing Systems*, 18:595–602.
- [18] Jolivet, R., Rauch, A., Lüscher, H.-R., and Gerstner, W. (2006b). Predicting spike timing of neocortical pyramidal neurons by simple threshold models. *Journal of Computational Neuroscience*, 21(1):35–49.
- [19] Kistler, W. M., Gerstner, W., and Hemmen, J. L. V. (1997). Reduction of the Hodgkin-Huxley Equations to a Single-Variable Threshold Model. *Neural Computation*, 9(5):1015–1045.
- [20] Lumer, E. D. (2000). Effects of spike timing on winner-take-all competition in model cortical circuits. *Neural Computation*, 12(1):181–94.
- [21] Maass, W. (1996). On the Computational Power of Noisy Spiking Neurons. *Advances in Neural Information Processing Systems*, 8:211–217.
- [22] Maass, W. (1999). *Computing with Spiking Neurons*, chapter 2, pages 55–86. The MIT Press.
- [23] Maass, W. (2000a). Neural Computation with Winner-Take-All as the Only Nonlinear Operation. *Advances in Neural Information Processing Systems*, 12:293–299.
- [24] Maass, W. (2000b). On the Computational Power of Winner-Take-All. *Neural Computation*, 12(11):2519–35.

- [25] Maass, W. and Bishop, C. M., editors (1999). *Pulsed Neural Networks*. The MIT Press.
- [26] Markram, H., Toledo-Rodriguez, M., Wang, Y., Gupta, A., Silberberg, G., and Wu, C. (2004). Interneurons of the Neocortical Inhibitory System. *Nature reviews. Neuroscience*, 5(10):793–807.
- [27] Nessler, B., Pfeiffer, M., and Maass, W. (2009). STDP enables spiking neurons to detect hidden causes of their inputs. *Advances in Neural Information Processing Systems*, 22:1357–1365.
- [28] Nessler, B., Pfeiffer, M., and Maass, W. (2010). Bayesian Computation Emerges in Generic Cortical Microcircuits through Spike-Timing-Dependent Plasticity. (*in preparation*).
- [29] Oster, M., Douglas, R., and Liu, S.-C. (2009). Computation with Spikes in a Winner-Take-All Network. *Neural Computation*, 21(9):2437–65.
- [30] Purves, D., Brannon, E. M., Cabeza, R., Huettel, S. A., LaBar, K. S., Platt, M. L., and Woldorff, M. G. (2008). *Principles of Cognitive Neuroscience*. Sinauer Associates, Inc.
- [31] Rutishauser, U. and Douglas, R. J. (2009). State-dependent computation using coupled recurrent networks. *Neural Computation*, 21(2):478–509.
- [32] Sjöström, J. and Gerstner, W. (2010). Spike-timing dependent plasticity. *Scholarpedia*, 5(2):1362.
- [33] Stevens, C. F. and Zador, A. (1996). When is an Integrate-and-fire Neuron like a Poisson Neuron? *Advances in Neural Information Processing Systems*, 8.
- [34] Thomson, A. M., West, D. C., Wang, Y., and Peter, B. A. (2002). Synaptic connections and small circuits involving excitatory and inhibitory neurons in layers 2-5 of adult rat and cat neocortex: triple intracellular recordings and. *Cerebral Cortex*, 12(9):936–53.
- [35] Unger, M. (2010). Frameworks for the Analysis of Learning with Biological Plausible WTA Circuits. Seminar Project, Available on Request: unger@igi.tugraz.at.

- [36] Womack, M. and Khodakhah, K. (2002). Active contribution of dendrites to the tonic and trimodal patterns of activity in cerebellar Purkinje neurons. *The Journal of Neuroscience: The official Journal of the Society for Neuroscience*, 22(24):10603–12.
- [37] Yuille, A. L. and Grzywacz, N. M. (1989). A Winner-Take-All Mechanism Based on Presynaptic Inhibition Feedback. *Neural Computation*, 1(3):334–347.