Department of Physics and Astronomy
Heidelberg University

Bachelor thesis in Physics
submitted by

Quirinus Schwarzenböck

from Garmisch-Partenkirchen

2019

Towards Balanced Random Networks on the BrainScaleS I System

This Bachelor thesis has been carried out by

Quirinus Schwarzenböck

at the

Kirchhoff Institute for Physics

at

Heidelberg University

under the supervision of

Dr. Johannes Schemmel

Towards Balanced Random Networks on the BrainScaleS I System

Quirinus Schwarzenböck

Abstract

Under the premise that the cortical column network, developed by Potjans and Diesmann (2012) is currently being implemented on BrainScaleS I hardware, a mixed-signal system for the emulation of spiking neural networks, this thesis is taking a closer look on which effects are likely to occur when transferring a network from digital to analogue simulations. This is done by taking the neural network described in Brunel (2000), a network that has a close resemblance to some parts of the cortical column network, and investigate how it reacts to the transfer to leaky integrate-and-fire neurons with current and conductance based synapse models, to distortions on the neuron parameters, as they are expected to occur on hardware, and other possible obstacles one might be facing during this transition, like a limited range the synaptic time constant can be chosen from. Paving the road for a successful emulation of the column, this allows a first prediction how the network could behave on BrainScaleS I and which parameters have to be handled carefully during the transmission and for which parameters the network is showing stable behaviour, even if they are connected with some distortions.

Zusammenfassung

Mit Hinblick auf die aktuell laufende Implementierung des Cortical Column Netwerks (Potjans und Diesmann 2012) auf BrainScaleS I, einem Mixed-Signal System das Gepulste Neuronale Netze emuliert, beschäftigt sich diese Arbeit mit den Effekten die auftreten, wenn man Netzwerke die bisher digital simuliert wurden auf analoger Hardware laufen lässt. Um dies zu erreichen, wird das ursprünglich von Brunel (2000) beschriebene Netzwerk, ein Netzwerk das große Ähnlichkeit mit Teilen des Cortical Column Netzwerkes aufweist, auf den Übergang zu Leaky Intergate-And-Fire Neuronen mit strom- und spannungsbasierten Synapsen Modellen, den Einfluss den eine Verteilung der Neuronen Parameter auf das Verhalten des Netzwerks haben und andere Schwierigkeiten die bei dem Übergang auf die Hardware auftreten können, beispielsweise dem limitierten Bereich in dem die synaptische Zeitkonstante gewählt werden kann untersucht. Das erlaubt eine erste Einschätzung wie sich das Cortical Column Netzwerk auf der Hardware verhalten wird, in welchen Fällen sich das Netzwerk stabil verhalten wird oder in welchen Fällen Unterschiede zu erwarten sind.

Contents

Chapter 1

Introduction

The human brain is known to be a powerful computing unit. It demonstrates quick learning capabilities and does so while exhibiting amazing robustness against hardware defects (Piccinini and Bahar 2013). These points justify the efforts that is being made to understand how the brain works in general on the one hand and to check whether these mechanisms can be incorporated into future computer hardware.
The field of research following up on this idea is called neuromorphic computing. Among other things, neuromorphic computing uses *Spiking Neural Networks* (SSN) as a base for computations. SSNs can be simulated either digitally, an approach taken by *Manchester University* with Spinnaker (Furber et al. 2014), or one can take a more radical approach and design hardware chips that are simulating neurons analogously and run the networks on these chips. The latter way, known as physical modelling, has been chosen by Electronic Vision(s) Group at Heidelberg University with their system BrainScaleS (Schemmel et al. 2010).
There are however some differences when running SSNs on software or on hardware. Some problems are affecting both systems. The limited bandwidth for example between neurons can lead to data packages being dropped instead of transmitted if too much spikes have to be transmitted at the same time. But there are also problems specific to the simulation type. When simulating neural networks on software, calculations are done numerically and errors in the simulation are due to numeric approximations. On hardware on the other hand, parameters have to be set on said hardware. This will likely lead to some variation of the parameters in the simulated network, since it is not possible to manufacture every single part of a chip in exactly the same way. One way to counteract these imperfections is to calibrate the individual hardware parts, which does help but can not make the chip perfect.

One possible application of BrainScaleS is a running version of the Cortical Column network, a well-known network in neuromorphic computing that has been used by a variety of groups as a benchmark, including Spinnaker (Sacha J. van Albada et al. 2018).
The Cortical Column is build out of several groups of populations of neurons. One of these groups has a close resemblance to a network described and analysed in Brunel (2000). This network shall be the basis of this thesis, with the goal to investigate the effect of applying variation to different parameters and simulating the model with different neuron models.

Chapter 2

Theory and Methods

Biologically speaking, neurons are cells with an electrically excitable cell membrane. The potential difference to their surroundings is called *membrane potential*. Under certain circumstances, for example when the membrane potential reaches a certain value, they are capable of producing short peaks in the membrane potential, so called *spikes*. They are connected to other neurons via synapses, which are transmitting these spikes between neurons (Brandes, Lang, and R. F. Schmidt 2019).
Spikes arriving at neurons are activating a wide variety of ion channels, that are responsible for changing the neurons membrane potential. Although these processes have been a subject of research for quite some time now, they are still computationally expensive and in some cases only solvable numerically. The different ion channels can roughly be matched with one of two categories: inhibitory, which will lower the membrane potential, and excitatory, that will cause a rise in the membrane potential (Gerstner, Kistler, et al. 2014).
To simplify this process one tries to mimic a neurons behaviour with more simple models, in this thesis the *leaky integrate-and-fire* (LIF) model, that will be explained in section 2.1, with a more detailed look on three subtypes, that are used in this thesis.
While it is still part of ongoing research how information between neurons is transmitted exactly (Gerstner, Kreiter, et al. 1997), the LIF model is based on the assumption that the timing of spikes is a crucial component carrying information when working with neurons. It therefore aims at reproducing the production of spikes in a neuron, a so called spiketrain. With this spiketrain, one can not only look at the individual spikes, but also at the rates that spikes are emitted at, how regular neurons are emitting spikes and if there is some correlation between the timing of the spikes. The important quantities for this thesis are discussed in section 2.3.

2.1 Leaky Integrate-and-Fire Model

A simplified model to describe the membrane potential $u(t)$ of a neuron as a function of time is a *leaky integrate-and-fire* (LIF) neuron. It consists of a capacitor C_m, a resistor that is characterised by its conductivity g_L and a so called *resting potential* V_L, all wired in a closed circuit (see figure 2.1). The simulated membrane potential is then defined as the potential of the capacitor.
The neuron will produce a spike when the membrane potential is above the *threshold value*

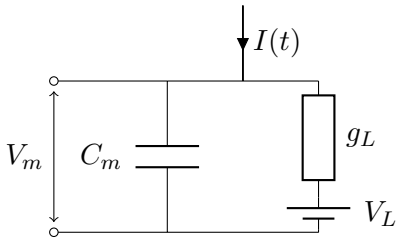


Figure 2.1 General circuit diagram for a LIF neuron with input current $I(t) = I_{\text{ext}}(t)+I_{\text{syn}}$.

Θ. While the membrane potential of the neuron is below mentioned threshold, its behaviour is characterised by (Gerstner, Kistler, et al. 2014)

$$C_m \frac{\mathrm{d}u}{\mathrm{d}t} = g_L(V_L - u) + I_{\text{syn}} + I_{\text{ext}}. \tag{2.1}$$

The possible input currents are I_{syn}, a current dependent on the spikes the neuron is receiving and I_{ext}, a current that is externally provided and therefore independent of the network and its spikes. Without any input currents ($I_{\text{syn}} = I_{\text{ext}} = 0$) the membrane potential will decay to the resting potential.
An additional definition that can be useful to characterise a LIF neuron is the *membrane time constant* τ_m given by

$$\tau_m = \frac{C_m}{g_L}. \tag{2.2}$$

In case the membrane potential reaches its threshold at time t_s the neuron spikes at $t = t_s$. After a spike the neuron is reset to its reset potential $V_r < \Theta$ and remains there for the duration of its *refractory time* τ_{ref}:

$$u(t) = V_r, \quad \text{for } t_s < t < t_s + \tau_{ref}. \tag{2.3}$$

To connect two neurons one uses synapses. Synapses transmit information only in one direction, from the presynaptic neuron to the postsynaptic neuron. With this model the only information transmitted between connected neurons is the emission of a spike, along with the information whether the neuron emitting the spike is inhibitory or excitatory. The time that passes between a spike being emitted by the presynaptic neuron and received by the postsynaptic neuron is called the *synaptic delay* τ_{syn} and is specific for the respective synapse. The other parameter characterising a synapse is the *weight* ω, determining how much influence a spike will have on a neuron (Brunel and Rossum 2007).
There are several ways I_{syn} can be described. For this thesis δ-input, current, and conductance based LIF neurons are being used and described in the following paragraphs.

2.1.1 δ-input based LIF neurons

δ-input based LIF neurons can actually be seen as a special case of the CUBA LIF neurons, which are explained in the next section. However, they can be described more easily than current based neurons and demonstrate a very particular behaviour, which makes it useful to have a more detailed look at them.
δ-input based neurons need the least parameters to be described. The idea is that a spike arriving at the neuron will instantly add a certain potential Δu to the membrane potential u. This allows equation 2.1 to be rewritten as formula 2.4, where the index k characterises the summation over all presynaptic neurons with the weights Δu^k, and the index s the summation over all the spikes of a neuron k:

$$\tau_m \frac{\mathrm{d}u}{\mathrm{d}t} = (V_L - u) + \sum_k \sum_s \Delta u_k \delta(t - t_s^k). \tag{2.4}$$

Since the potential difference is instantaneous and fixed, the capacity together with the conductivity can be combined to one parameter τ_m according to equation 2.2. Describing how fast the membrane potential is exponentially decaying towards the resting potential, the membrane time constant gives in this case a more direct insight to the neurons behaviour than the capacity or the conductance (Brunel 2000).

2.1.2 Current based LIF neurons

Current based (CUBA) LIF neurons are mathematically more easily describable than conductance based neurons, since the effect an arriving spike is having on the membrane potential is independent of the state of the neuron.
The idea of current based synapses is, that a spike s from neuron k arriving at the neuron will trigger a current to the neuron, scaled by the weight of the synapse ω^k. Assuming that the kernel does not have a unit of its own, for this model the weight ω has the unit Ampère. The incoming current is described by a kernel $\varepsilon_n(t)$, specific to the neuron itself and whether the arriving spike is inhibitory ($n = \mathrm{inh}$) or excitatory ($n = \mathrm{exc}$). This gives the synaptic input the form:

$$I_{\mathrm{syn}} = \sum_k \sum_s \omega_k \varepsilon_n(t - t_{k,s}). \tag{2.5}$$

Some advantages of this model are the already mentioned independence of I_{syn} from the state of the network and that the membrane time constant (see equation 2.2) characterises the decrease of the membrane potential as well (Petrovici, Bill, and Hartl 2017).

2.1.3 Conductance based LIF Neurons

Conductance based (COBA) LIF neurons use the approach that is the closest one of the three depicted models to the original ion channels. The idea is to connect additional pairs of potentials and controllable resistors, wired in series, parallel to the capacitor and change the membrane potential by regulating the conductivity of the additional resistors (Gerstner, Kistler, et al. 2014).
Explicitly one adds one series of potential and resistor for excitatory spikes and one for inhibitory spikes, as can bee seen in figure 2.2. The synaptic input current is then given by equation 2.6, the behaviour of the conductance by equation 2.7. The conductances of the additional resistors are calculated by a kernel specific to the synapse $\varepsilon_n(t)$ and whether the triggering spike was emitted by an inhibitory ($n = \text{inh}$) or excitatory ($n = \text{exc}$) neuron. The conductance is scaled by a weight ω^k determined by the synapse connecting the postsynaptic neuron to the presynaptic neuron k. Keeping the assumption that the kernel does not have a unit, for this model the ω has the unit Siemens:

$$I_{\text{syn}} = g_{\text{exc}}(t)(V_{\text{exc}} - u) + g_{\text{inh}}(V_{\text{inh}} - u), \tag{2.6}$$

$$g_n(t) = \sum_s \sum_k \omega_k \varepsilon_n(t - t_{k,s}),\ n \in \{\text{exc},\ \text{inh}\}. \tag{2.7}$$

As with previous models the membrane time constant can still be defined by equation 2.2, but with this approach the quantity loses its physical meaning. To characterise the decay of the membrane potential with this model, the *effective membrane time* τ_{eff} is introduced. It still characterises an exponential decay of the membrane potential, but is calculated according to:

$$\tau_{eff} = \frac{C_m}{g_L + g_e + g_i}. \tag{2.8}$$

This directly leads to the properties and more importantly differences of this model to the previously explained and more alike models. The dynamic range is limited by the excitatory and inhibitory potentials, while so far it has been theoretically unlimited. The effective membrane time is not constant anymore, but depends on arriving spikes. The change in membrane potential for arriving spikes is also not constant anymore but changes with the state of the neurons membrane potential. Considering all of these aspects the transition from CUBA to COBA LIF neurons can not necessarily expected to be trivial.

2.2 Network Architecture

The network investigated for this thesis is randomly and sparsely connected. It was adopted from Brunel (2000), where it was described and mathematically analysed.

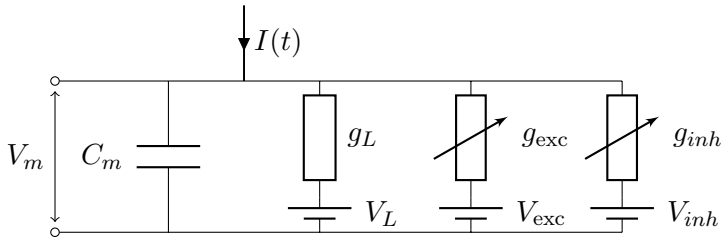


Figure 2.2 Conductance based LIF neuron with potentials V_i and adjustable conductances $g_n(t)$, $n \in \{\text{exc}, \text{inh}\}$.

The network consists of N neurons, $N_{\text{exc}} = 0.8N$ of which are excitatory and $N_{\text{inh}} = 0.2N$ inhibitory. The used neuron model is the δ-input based LIF neuron described in section 2.1.1. Each neuron is connected to $C = C_{\text{exc}} + C_{\text{inh}}$ presynaptic neurons, with $C_{\text{exc}} = \varepsilon N_{\text{exc}}$ excitatory and $C_{\text{inh}} = \varepsilon N_{inh}$ inhibitory connections. For this thesis $\varepsilon = 0.1$.
The weight of these connections is determined by the parameters J_{exc} and $J_{\text{inh}} = -gJ_{\text{exc}}$, where J_{exc} is the potential difference added to the membrane potential for an excitatory spike, J_{inh} for an inhibitory spike. The parameter g determines how J_{exc} and J_{inh} relate to one another. It is one of the two parameters used to influence the networks behaviour and chosen between zero and eight for this thesis, as in Brunel (2000).
Additionally each neuron is connected to as many external spike sources as it is connected to excitatory neurons ($C_{\text{ext}} = C_{\text{exc}}$ external spike sources). All of them are excitatory and their synaptic weights are equal to the synaptic weights of excitatory spikes inside the network ($J_{\text{ext}} = J_{\text{exc}}$). These sources are sending spikes according to a Poisson distribution with a mean frequency ν_{ext} that is calculated by equation 2.9. To calculate ν_{ext} one needs the *threshold frequency* ν_{thr}, which is defined as the minimal regular frequency a neuron needs to receive excitatory spikes at, that will cause a neuron to reach its threshold in the limit $t \to \infty$, and η, the second parameter used to influence the network:

$$\nu_{\text{ext}} = \eta \nu_{\text{thr}}, \tag{2.9}$$

$$\nu_{\text{thr}} = \frac{\Theta}{J_{\text{exc}} C_{\text{exc}} \tau_m}. \tag{2.10}$$

The final parameter needed to characterise the network is the synaptic delay D that will be chosen between 1.5 ms and 3.0 ms. A summary of all the networks parameters can be found in table 2.1.

2.3 Network Behaviour Analysis

When analysing the behaviour of the networks in this thesis there are two aspects that are taken into consideration. On the one hand one looks at the regularity of individual neurons, describing how constant their firing rate is with respect to time. The mathemat-

Symbol	Value	Description
Θ	20 mV	Threshold of the neurons
V_{rest}	0 mV	Resting potential of the neurons
V_{reset}	10 mV	Reset potential
J_{exc}	0.1 mV	Weight for excitatory spikes
τ_m	20 ms	Time constant of a neuron membrane
τ_{ref}	2 ms	Refractory time of the neurons
D	1.5 ms to 3 ms	Synaptic delay
η	1 to 4	Parameter determining the external frequency
g	0 to 8	Relation between weight for excitatory and inhibitory spikes

Table 2.1 Summary of the parameters of the original network with δ-input based neurons.

ical background is presented in section 2.3.1. On the other hand one is interested in how the neurons are correlated to each other. This is further explained in section 2.3.2. The techniques introduced in the following paragraphs differ from the approach to classify the network used in Brunel (2000), where the networks are looked at analytically and the classification of the networks, or more exactly the transition from one state to another, is given by *Hopf bifurcations*.

2.3.1 Neuron Behaviour

The first quantity introduced to analyse the behaviour of a single neuron is the *interspike interval* (ISI), defined as the time passed between two spikes of a neuron. To figure out how *regularly* a neuron is firing the *coefficient of variation* (CV), the ratio of the standard deviation to the mean (see equation 2.11) is used:

$$CV(X) = \frac{\sigma(X)}{\mu(X)}. \tag{2.11}$$

One advantage of this quantity is its normalisation to its mean. This allows a more meaningful comparison between data sets with a wider variety of means. The disadvantage that comes with this normalisation applies to data sets with a mean close to zero. In these cases the CV is very sensitive to slight variations in the mean of the data sets, making a comparison of their CV unhelpful.
The CV of the ISI of a single neuron then describes the regularity of a neuron, the mean of all of the ISIs, CV_{ISI}, in a network is used to characterise the neuron behaviour in a network.

2.3.2 Global Behaviour

The behaviour of all neurons combined, called global behaviour as of now, describes how synchronous neurons are firing. In Brunel (2000) a networks *synchronicity* is determined by its *common instantaneous firing rate* $\nu(t)$. The common instantaneous firing rate is defined by the probability $\nu(t)\mathrm{d}t$ of each neuron to emit a spike in the time interval t to $t+\mathrm{d}t$. A constant instantaneous firing rate $\nu(t) = \nu$ means that all neurons are firing independent of each other and are therefore uncorrelated and the network behaves *asynchronous*. A *synchronous* network on the other hand is defined by a instantaneous firing rate $\nu(t)$ varying over time.
To classify networks, an approach used for similar networks, for example in Potjans and Diesmann (2012) was employed: a histogram of all spikes in the network is created and the CV (see equation 2.11) of the bin heights is calculated. Employing the definition above of synchronicity, the bins of a histogram of an asynchronous network can be expected to be about equal height, while the bins of a synchronous network will display a lot more variation. To account for differences in the the overall network activity, i. e. the mean frequency a spike is emitted in the network, the CV seems to be a well equipped measure for the global behaviour, as of now referred to as CV_{G}.

2.4 Network Behaviour

With two states for each, the neuron and global behaviour (see section 2.3), there is a total of four states the network can be in. According to Brunel (2000), these states are:

- In the *synchronous regular* (SR) state almost all neurons are firing simultaneously in a few clusters. These clusters behave as oscillators, that can be seen in the global activity. CV_{ISI} is small, CV_{G} high.

- The *asynchronous regular* (AR) state presents with very regularly firing neurons as well, but the neurons are firing independent of each other, leading to small CV_{G} and CV_{ISI} values.

- *Asynchronous irregular* (AI) states manifest with stationary global activity and irregularly firing neurons. CV_{ISI} is high, CV_{G} is small.

- *Synchronous Irregular* (SI) have oscillatory global activity and strongly irregular behaving neurons. Both CV_{ISI} and CV_{G} are high.
  For high external frequencies the global oscillations are faster than for low external frequencies. While this can qualitatively seen in the simulation data, it has not been taken into consideration in the analysis of the simulated networks.

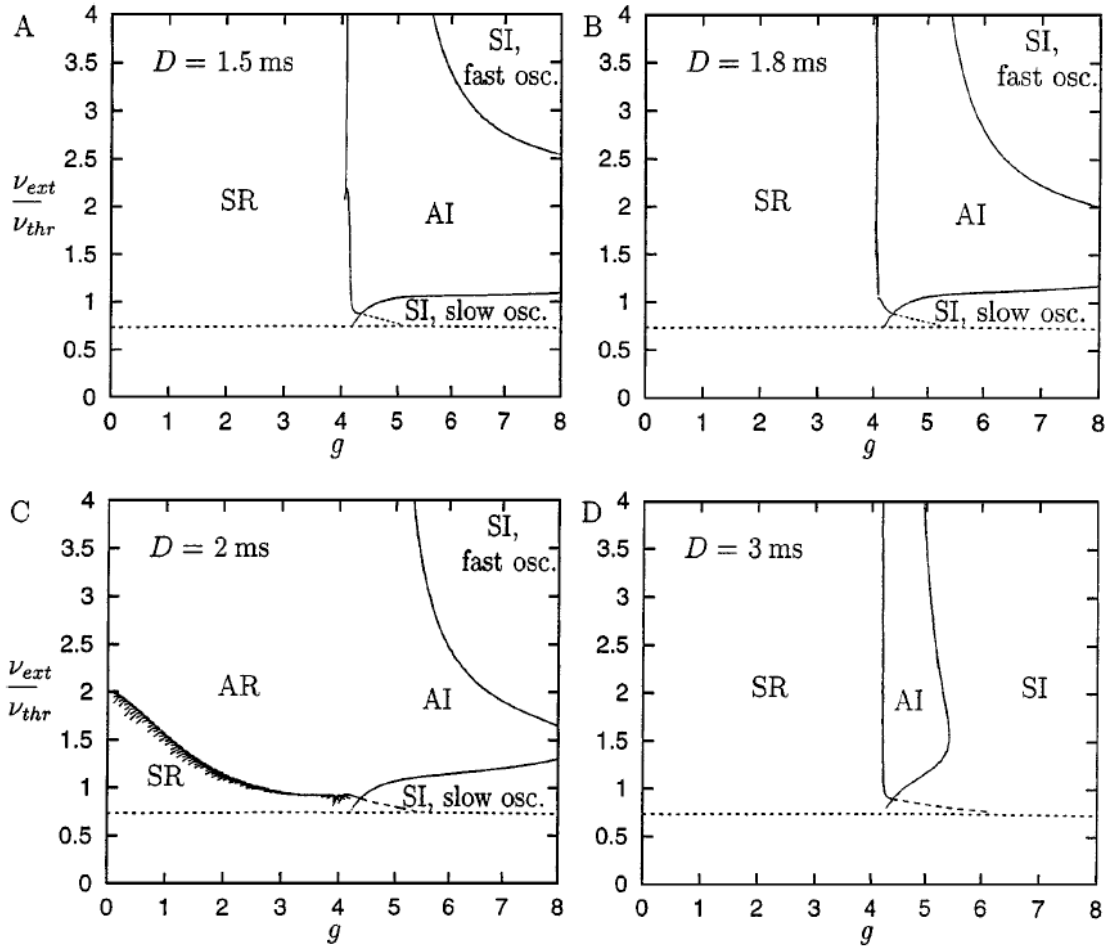


Figure 2.3 The theoretically expected network behaviour for different delays D as functions of the parameters g and η. The different regions are labelled as introduced in section 2.4. The continuous lines are Hopf bifurcations between the different states.

Under which circumstance these states are theoretically expected for a network build according to the architecture described in section 2.2 are pictured in figure 2.3. An example for each of these states is depicted in figures 3.3 and 3.4.

2.5 BrainScaleS System

As this thesis investigates possible differences in the behaviour of a network, when it is simulated on classical computers versus when it is emulated on hardware, namely the BrainScaleS system, the latter one is shortly introduced here.
The BrainScaleS system, developed at Kirchhoff-Institute for physics at Heidelberg University by the Electronic Vision(s) Group, is a system designed for the emulation of neuromorphic processes. The crucial part of the systems are wafers, each consisting of 384 *High*

Input Count Analogue Neural Network (HICANN) chips, one of which contain 512 neurons and 220 synapses for each neuron (Schemmel et al. 2010).
The emulation is done by a mixed-signal chip, the analogue part of which is responsible for emulating the membrane potential of a neuron and the synapses leading to it, based on the *Adaptive Exponential* (AdEx) LIF neuron model, a simplified version of the Hodgkin-Huxley-Model (Hodgkin and Huxley 1952), developed by Brette and Gerstner (2005). The digital part of the chip is responsible for deciding which neuron receives spikes from which other neuron. With appropriately chosen capacities and conductances (compare to equation 2.2), this hardware setup one second of a biological network can be emulated 10.000 times faster.

Chapter 3

Experiments and Results

The first part of this thesis focuses on simulating the network described in section 2.2 with CUBA LIF neurons and reproducing results from Brunel (2000) (see section 3.1.1). The network is then investigated on how it reacts to variation in neuron and synapse parameters, their initial membrane potential in section 3.2.2, as well as the effect that is caused by using COBA LIF neurons when simulating the network in section 3.3. The final step is to prolong the time it takes for the total change in membrane potential to happen in section 3.5. The final step is the transition to COBA LIF neurons in section 3.3. All of these effects are discussed with respect to possible problems one will be facing when implementing similar networks to the one in this thesis on BrainScaleS I.
The network is implemented using PyNN (Davison et al. 2009), a language used for implementing neural networks on several simulating engines and neuromorphic hardware (*PyNN: documentation* 2019). The simulations for this thesis use the simulation engine NEST (Peyser et al. 2017).
Besides the neuron models, the only difference to section 2.2 incorporated in the implementation of the network is that all external input a neuron receives is coming from only one source with mean frequency $\nu_{\text{ext, sim}} = C_{\text{ext}}\nu_{\text{ext}}$. Since the input is still generated to have a Poisson distribution this matches the input from combined multiple sources, that will have in total a Poisson profile if they start with one in the first place (Grimmett and Welsh 2014). It is however more efficient in computing time to reduce the sources for each neuron to one random number generator.

3.1 CUBA LIF Neuron Network

For the transition from the original network (see section 2.2) to CUBA LIF neurons, neurons with an exponential kernel are used:

$$\varepsilon_n(t) = \varepsilon(t) = \exp\left(-\frac{t}{\tau_{syn}}\right),\ n = \{\text{exc}, \text{inh}\}. \tag{3.1}$$

This kernel has been chosen, because it is the fastest decreasing kernel supported by PyNN, an important aspect when mimicking the behaviour of the δ-input based neurons. Additionally the COBA LIF neuron model supported by BrainScaleS hardware also uses an exponential kernel.
With this kernel, the synaptic time constant τ_{syn} is introduced, describing how fast the kernel, and therefore the current to the neuron, is decreasing with time. As a result of

this, the behaviour of a δ-input based LIF neuron will be most closely reproduced with small synaptic time constants.
To calculate the weight, that causes a specific change Δu in the membrane potential of the neuron, the equation describing the membrane potential of a CUBA LIF neuron with kernel 3.1 comes in handy (taken from Rudolph-Lilith, Dubois, and Destexhe (2012)):

$$u(t) = V_L + \sum_s \sum_k \Theta(t - t_s^k) \frac{\tau_m \tau_{syn,k} \omega_k}{C_m(\tau_m - \tau_{syn,k})} \left(\exp\left(-\frac{t - t_{k,s}}{\tau_m}\right) - \exp\left(-\frac{t - t_{k,s}}{\tau_{syn,k}}\right) \right). \tag{3.2}$$

Since a small synaptic time constant is necessary for mentioned reasons, the assumption $\tau_{syn} \ll \tau_m$, together with equation 3.2 leads to the following approximation for the weight ω as a function of Δu:

$$\Delta u \approx \frac{\tau_{syn}\omega}{C_m}, \tag{3.3}$$

$$\Rightarrow \omega \approx \frac{\Delta u C_m}{\tau_{syn}}, \text{ for } \tau_{syn} \ll \tau_m. \tag{3.4}$$

With the membrane time constant $\tau_\mathrm{m} = 20.0\,\mathrm{ms}$ and the synaptic time constant set to $\tau_\mathrm{syn} = 0.01\,\mathrm{ms}$, the weight that causes a change in membrane potential of $\Delta u = 0.10\,\mathrm{mV}$ should be about $\omega = 0.01\,\mathrm{nA}$. This has been experimentally tested to be accurate to two decimal places by simulating only one excitatory spike arriving at a neuron and reading ot the change in maximal change in membrane potential.
The synaptic time constant is set to $\tau_\mathrm{syn} = 0.01\,\mathrm{ms}$, since this allows almost the complete change in membrane potential to happen within one simulation time step ($\mathrm{d}t = 0.1\,\mathrm{ms}$). The capacitance of the capacitor of the LIF neuron is set to $1.0\,\mathrm{pF}$. Since the particular value of the capacitor does not influence the networks behaviour, because synaptic weight scales with it according to equation 3.4, it has been chosen from one of the examples available on the PyNN webpage (*PyNN: documentation* 2019). An overview of the parameters of the network can be found in 2.1, the additional parameters needed for the simulation of the network with CUBA LIF neurons are listed in table 3.1. The parameters for the simulation that have not yet been mentioned are discussed in section 3.1.2.

3.1.1 Reproduction of existing results

As a reference that the network is working properly the simulation from Brunel (2000), see figure 3.1, was recreated, see figure 3.2. How the parameters needed for the analysis of the network have been chosen is discussed in section 3.1.2
Looking at the results of the simulation, starting with the global behaviour (see figure 3.2a) one can see that, just as in figure 3.1, for $0.0 < g < 3.0$ the network is behaving synchronous for all investigated η values. In this area in plot 3.2a some spots are clearly

Symbol	Value	Description
C_m	1.0 pF	Capacitance of the neurons capacitor
τ_{syn}	0.01 ms	Synaptic time constant
g_L	5.0 pS	Conductance to the leak potentials
ω	0.01 nA	Calculated synaptic weight
dt	0.1 ms	Simulation time step
T	2000 ms	Duration of the simulation

Table 3.1 Additional parameters used for the simulation of the network with CUBA LIF neurons. The network parameters can be found in 2.1.

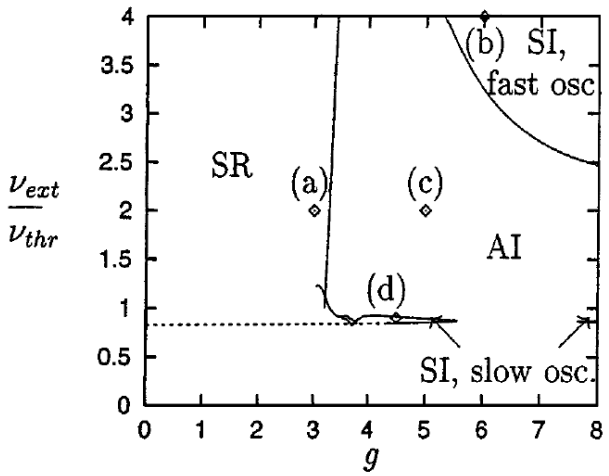
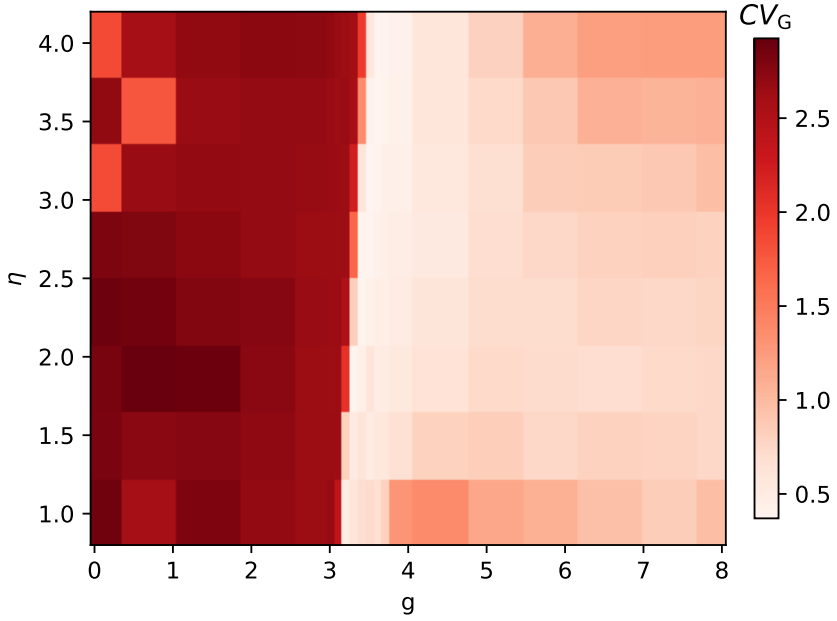


Figure 3.1 The simulated network behaviour by Brunel (2000) as functions of the parameters g and η.

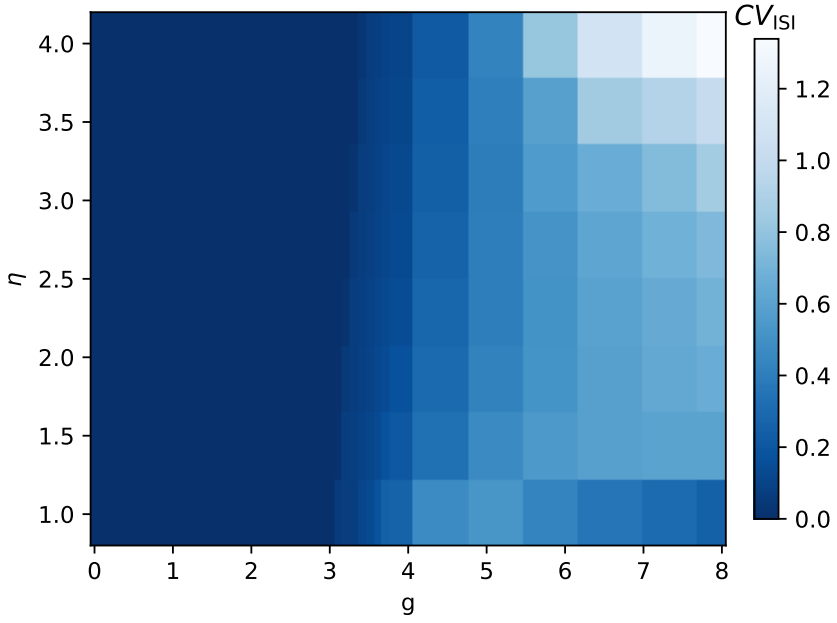
brighter than other ones. This can be explained by the number of clusters that are built by the neurons in the SR state. For a network in the SR state with two clusters, an example can be seen in 3.3a, the CV_{G} value is higher than for a network with three clusters, which is due to the calculation of te CV_{G} value. The CV_{G} value for all parameter sets is $CV_{\mathrm{G}} > 2.0$. It is also visible that in this area some spots are clearly brighter than For $3.0 < g < 3.7$ there is a clear transition from synchronous behaviour to asynchronous behaviour, the CV_{G} value falls to $CV_{\mathrm{G}} < 0.8$. The transition takes place at $g = 3.0$ for $\eta = 1$ and then shifts, linearly with respect to η, to $g = 3.7$ at $\eta = 4.0$.
This effect is also visible when looking at the neuron behaviour, which changes from regular firing states, where all states have $CV_{\mathrm{ISI}} < 0.5$ to irregular firing ones along the same transition line. The contrast of this transition is however much more subtle (see plot 3.2), since the value of CV_{ISI} is increasing with further increase of g.
While the neuron behaviour keeps being irregular with a further increase in g for all η values, since CV_{ISI} is only increasing further, the global behaviour is more interesting. In the upper right corner, i e. for $g > 5.5$ and $\eta > 3.0$, as well as for parameter sets with $\eta = 1.0$ and $3.8 \leq g \leq 5$, the CV_{G} value increases to $CV_{\mathrm{G}} > 1.2$, making the network again synchronous. Together with the irregular neuron behaviour, the network is in these areas in SI regimes.
In these SI regimes the values for CV_{G} are not as high as in the SR regime, which is a result of the fact that the neurons are not firing in clusters any more, but in a irregular way that leads to a oscillatory behaviour in the global activity, making the CV_{G} value smaller. This can be seen when comparing the global behaviour between figures 3.3b (SI network) and 3.3a (SR network). While this means that the CV_{G} values are not as high as in the SR regime, it is still clearly distinguishable from the asynchronous behaviour. For a better understanding of the states the network can be in according to section 2.4, in figure 3.3 and 3.4 each of the networks states is visualised, by plotting the spiking times of some randomly selected neurons to visualise the neuron behaviour as well as the histogram the histogram of the global behaviour that depicts the global behaviour.

3.1.2 Simulation and Characterisation Parameters

This section takes a look on how the parameters for the simulation and the analysis methods have been chosen.
Each network parametrisation is simulated for $T = 2000\,\mathrm{ms}$ with simulation time steps of $d_t = 0.1\,\mathrm{ms}$. The simulation time accounts for the time the network needs to reach its final state in which the individual neuron behaviour and global behaviour would not change anymore. This process, as spot checks for several parameter sets, that are covering all four network states, have revealed, could take up to $700\,\mathrm{ms}$. To account for eventual outliers a time buffer was added prior to the data that was used for the network analysis which started at simulated time $t = 1000\,\mathrm{ms}$ and ended with the end of the simulation at

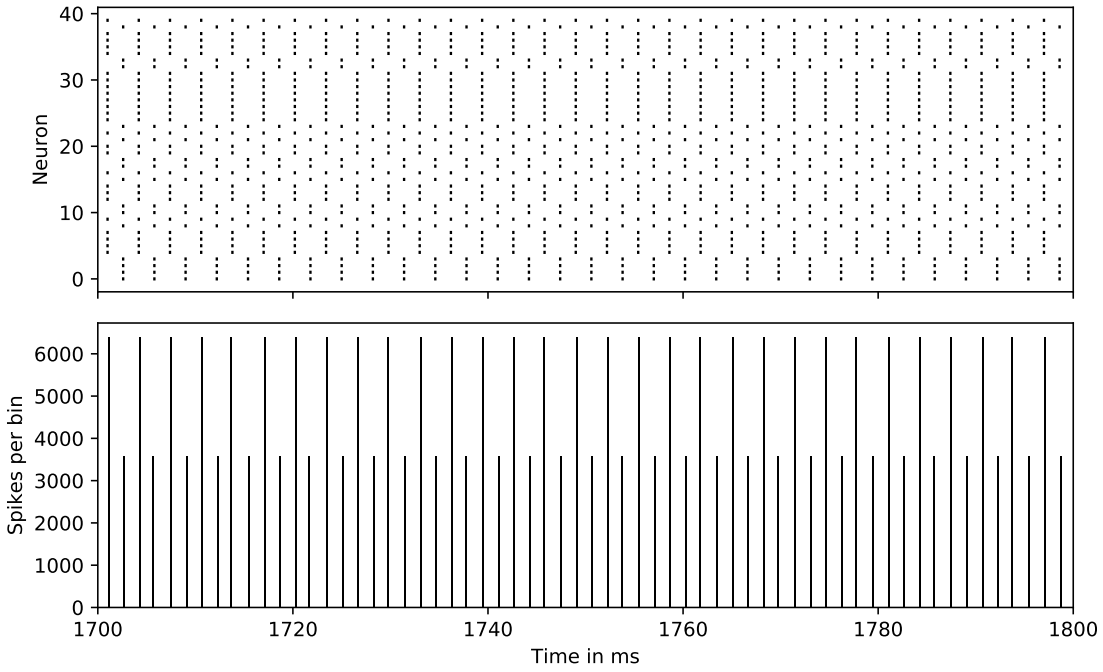


(a) Plot of the characteristic of the global activity CV_G.

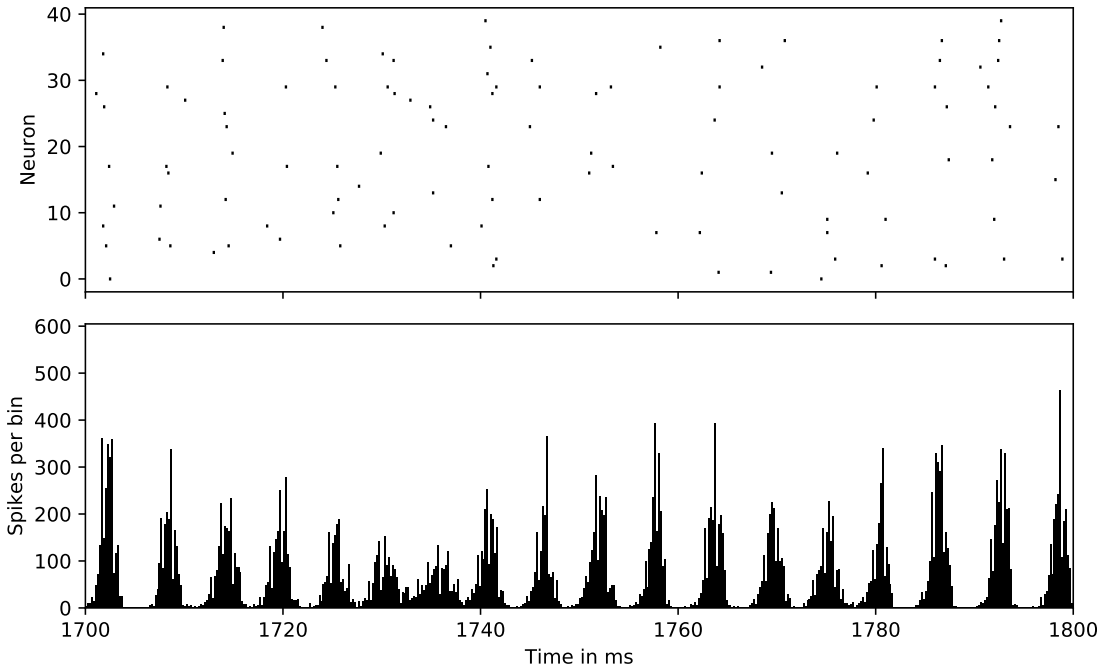


(b) Plot of the characteristic of the neuron activity CV_{ISI}.

Figure 3.2 Plot of the characteristics of the network. In the area between $2.9 \leq g \leq 3.8$ more data points have been recorded than for the other parts of the plot, since this allows the depiction of the transition line between the different network states more easily and recreating this transition line is one of the more interesting parts of the plot.

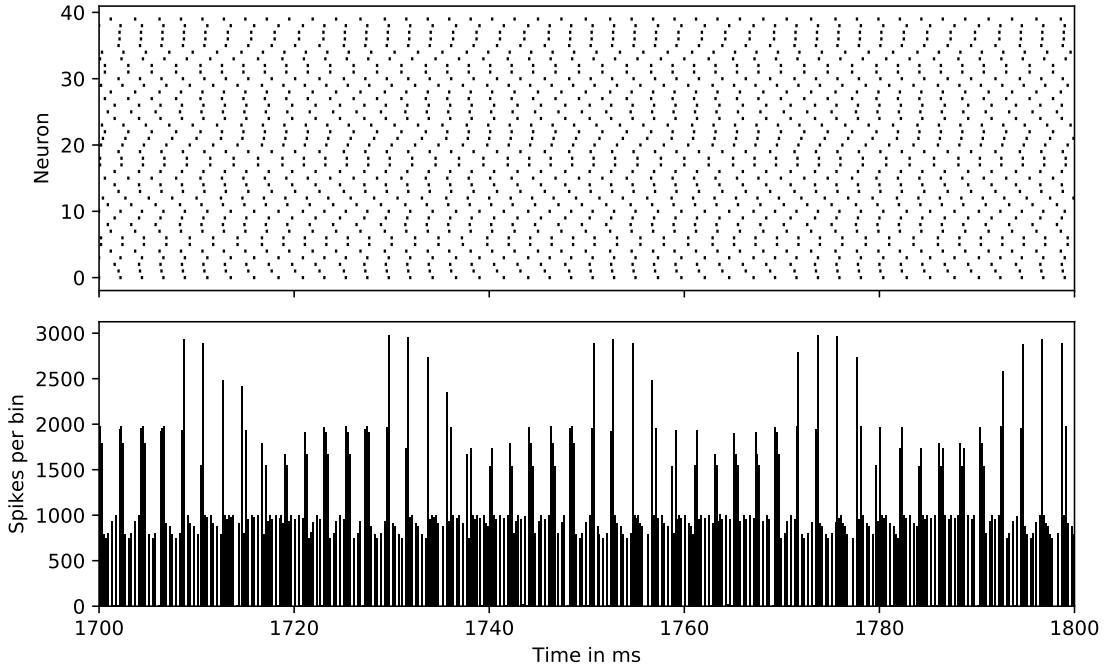


(a) Network in the Synchronous Regular (SR) state with $CV_{\mathrm{ISI}} = 0.00$ and $CV_{\mathrm{G}} = 2.76$.

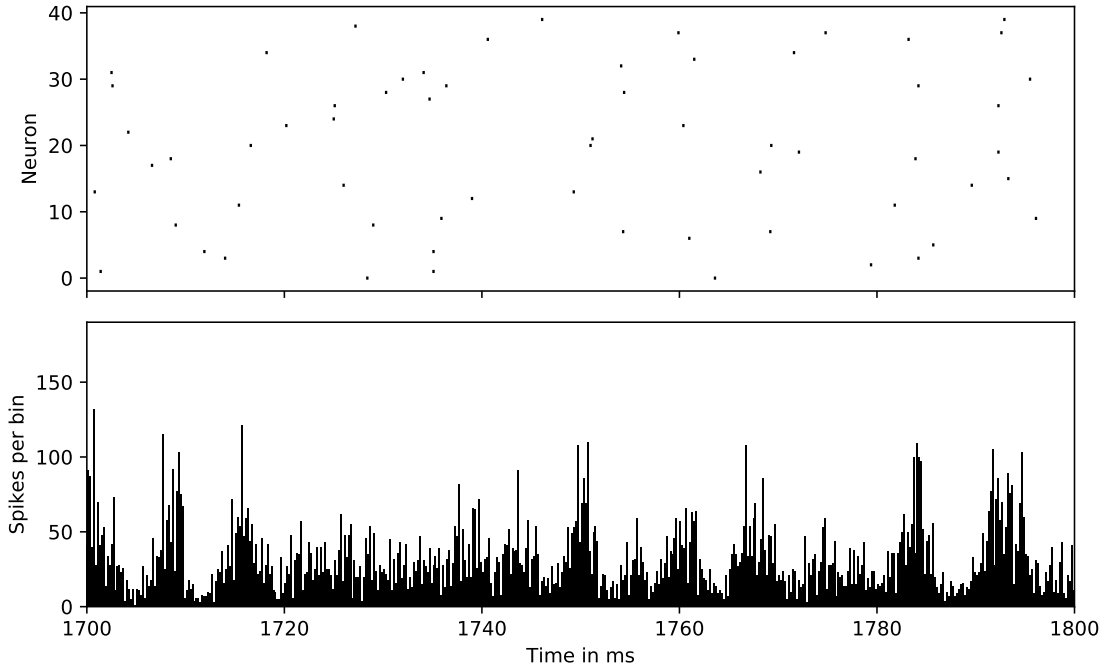


(b) Network in the Synchronous Irregular (SI) state with $CV_{\mathrm{ISI}} = 1.34$ and $CV_{\mathrm{G}} = 1.24$.

Figure 3.3 Examples for described synchronous states the investigated network can be in. The upper part of each plot pictures the spiketimes of 40 randomly selected neurons, while the lower part is the histogram also used for the global analysis, calculated as described in the text. For both figures, 3.3a ($g = 2.2$, $\eta = 2.2$), and 3.3b ($g = 8.0$, $\eta = 4.0$), a delay of $D = 1.5\,\mathrm{ms}$ has been used. Apart from these settings the network parameters are set according to tables 2.1 and 3.1.



(a) Network in the Asynchronous Regular (AR) state with $CV_{\mathrm{ISI}} = 0.59$ and $CV_{\mathrm{G}} = 0.71$.



(b) Network in the Asynchronous Irregular (AI) state with $CV_{\mathrm{ISI}} = 0.04$ and $CV_{\mathrm{G}} = 0.80$.

Figure 3.4 Examples for described asynchronous states the investigated network can be in. The upper part of each plot pictures the spiketimes of 40 randomly selected neurons, while the lower part is the histogram also used for the global analysis, calculated as described in the text. For figure 3.4b ($g = 6.5$, $\eta = 1.9$) a delay of $D = 1.5\,\mathrm{ms}$ has been used, for figure 3.4a ($g = 1.0$, $\eta = 3.5$) a delay of $D = 2\,\mathrm{ms}$. Apart from these settings the network parameters are set according to tables 2.1 and 3.1.

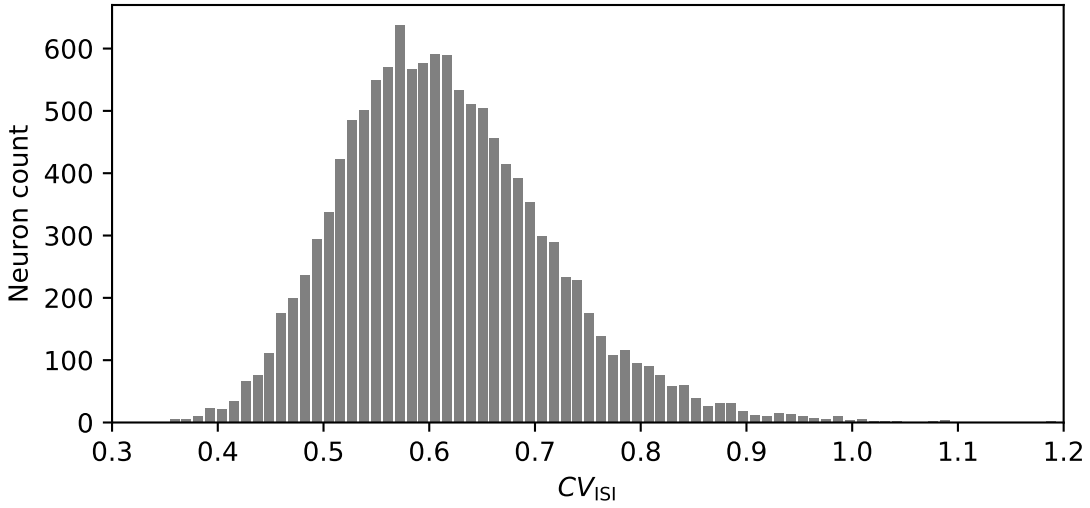
$t = 2000\,\mathrm{ms}$. Having $1000\,\mathrm{ms}$ to analyse the network proved to be enough time for the analysis methods to provide consistent results. This means that the network behaviour would lead to the same result as seen in the previous section. For the simulation time step smaller intervals between $d_t = 0.01\,\mathrm{ms}$ to $0.1\,\mathrm{ms}$ have been tested, which has lead to consistent results, so the simulation time step is set to $d_t = 0.1\,\mathrm{ms}$.
For the calculation of CV_G, a histogram with respect to the temporal development, of all the spikes is calculated (see section 2.3.2). Choosing a bin width $\mathrm{d}t_b$ for the histogram, several aspects have to be considered:

- The bin width should be a multiple of the simulation time step ($\mathrm{d}t = 0.1\,\mathrm{ms}$), in order to avoid artefacts when calculating the histogram.
- The bins have to be high enough to be distinguishable from background noise.
- The computation time.

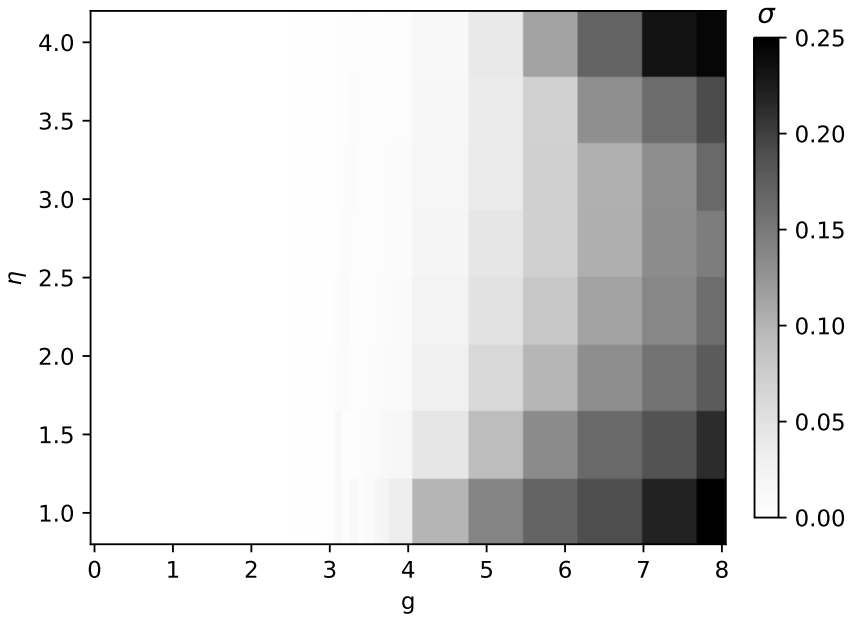
To take the first aspect into consideration, the analysis was tested for bin sizes between $0.1\,\mathrm{ms}$ and $1.0\,\mathrm{ms}$, with increments of $0.1\,\mathrm{ms}$. For bin sizes $\mathrm{d}t_b = 0.1\,\mathrm{ms}$ to $0.5\,\mathrm{ms}$ the produced results proofed to be consistent with one another. Looking at the second aspect, bin sizes $\mathrm{d}t_b \geq 0.2\,\mathrm{ms}$ lead for every network to bins with at least 100 events, which is considered to be enough. Since the computing time proved to be negligible, the bin size is set to $\mathrm{d}t_b = 0.2\,\mathrm{ms}$. With this analysis method the results of Brunel (2000) for the global behaviour can be reproduced (see section 3.1.1).
For the neuron analysis (see section 2.3.1) an important question that has to be asked is whether calculating the mean of the individual neurons CV_ISI is a legitimate quantity to characterise the network or not. This requires that the individual CV_ISI values are distributed around their mean, peaking at their mean value and that the standard deviation of this distribution is reasonably small. This has been investigated for all network states and in each case the distribution of the individual neurons CV_ISI values has been found to match the requirements. In figure 3.5a one of this distributions is being shown. The standard deviation for a network is shown in figure 3.5b. This makes it clear that the standard deviation σ of the individual neuron CV_ISI values is generally smaller than $\sigma < 0.25$. Therefore taking the mean of the individual neuron CV_ISI values should be a well suited measure.
In conclusion, both the analysis for the neuron behaviour and global behaviour is well suited for this network.

3.2 Distributed parameters

Having a network that is working as expected, the next goal is to investigate how it is going to react to varying parameters. So far all parameters have been set to one value for the complete network, without any deviation at all. Since BrainScaleS is a mixed-signal



(a) Exemplary distribution of individual neuron CV_{ISI}. The individual neurons CV_{ISI} are distributed around their mean ($\mu = 0.62$) with a standard deviation of $\sigma = 0.10$. The distribution was taken from an irregular regime with parameters $g = 6.0$, $\eta = 1.8$, $D = 1.5\,\mathrm{ms}$. Other parametrisations in the irregular regime lead to similar results.



(b) The standard deviation for a complete parameter sweep of a network with delay $D = 1.5\,\mathrm{ms}$.

Figure 3.5 Testing the neuron behaviour analysis.

chip, one has to expect some noise when setting network parameters, due to slight imperfections in the manufacturing process. This effect is approximated with a Gaussian parameter distribution that has a standard deviation of $10\,\%$ for parameters describing potentials and $20\,\%$ for temporal parameters (S. Schmitt: private communication). This leaves the deviations of capacitor of the neurons to be determined. The accuracy of the capacitors used in BrainScaleS hardware was investigated by D. Schmidt (2014) and was in every case more accurate than $5\,\%$ of the capacity. As a result, the distribution of the capacities has been conservatively approximated with a standard deviation of $5\,\%$ of the total capacity. The network is investigated towards the effect these variations cause individually and combined in section 3.2.2.
It is worth mentioning, that the delays of a network can not be configured, but are nevertheless present, depending on the routes that neurons are connected with on the chip. Therefore, it has been decided that the approach to simulate the effect by using a Gaussian distribution of the delay parameter with a standard deviation of $20\,\%$ should still be used for the investigation. Another parameter of the network that can not be set when emulating networks on BrainScaleS is the initial membrane potential. It is however still possible to influence the initial membrane potential, for example by giving a strong initial stimulus before the simulation starts or by only starting the simulation after enough time has passed that all neurons are at rest. To take this into account, the next section discusses the effect differently distributed initial neuron membrane potentials have on the behaviour of the network.

3.2.1 Distributed Initial Parameters

So far, everything was done with a constant initial membrane potential for all neurons $u(t = 0\,\mathrm{ms}) = 10\,\mathrm{mV}$. In this case the actual starting value should have no influence on the network since every neuron receives a Poissonian input, that raises their membrane potentials in a similar manner with only slight temporal variation.
Since BrainScaleS does only indirectly allow to influence the initial membrane potential, it is more interesting to look at the network while varying initial membrane potentials. For one, this was tested with a uniform distribution with varying width and with the difference that either all the membrane potentials have been smaller that the neurons threshold, but also for the case that a part of the population was initialised with membrane potentials above their threshold. This means that a part of the neuron population will emit a spike immediately after the simulation started.
This setup was done in a similar fashion with a normal distribution, i. e. with a standard deviation starting at a few percent and going up to thirty percent, and with the option that membrane potentials that would be initialised above the threshold are either redrawn from the distribution or allowed to stay that way. The explicitly tested variations are listed in table 3.2.

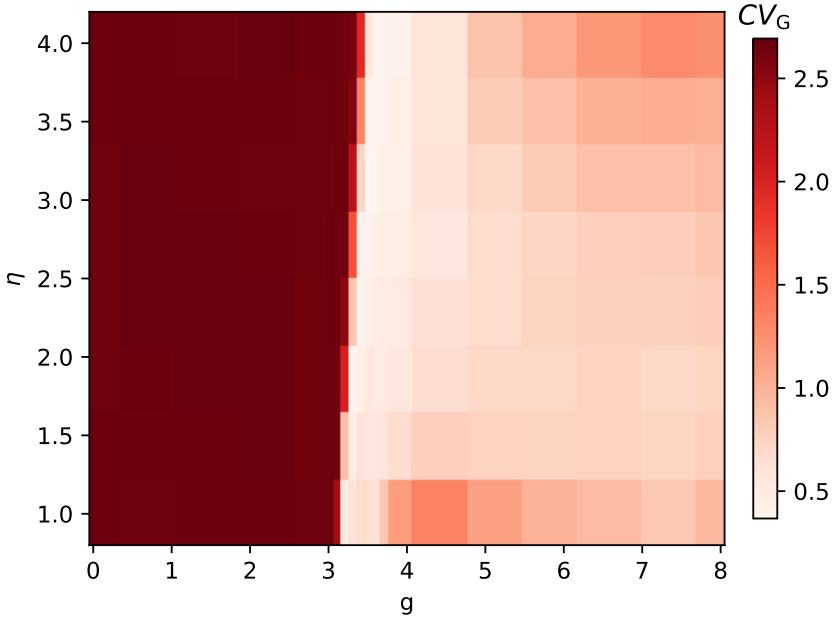
	Distribution	μ	σ	Range
A	uniform	-	-	$0\,\mathrm{mV}$ to $20\,\mathrm{mV}$
B	uniform	-	-	$-70\,\mathrm{mV}$ to $20\,\mathrm{mV}$
C	uniform	-	-	$-30\,\mathrm{mV}$ to $50\,\mathrm{mV}$
D	normal	$10\,\mathrm{mV}$	$4\,\mathrm{mV}$	-
E	normal	$20\,\mathrm{mV}$	$4\,\mathrm{mV}$	-
F	normal	$0\,\mathrm{mV}$	$20\,\mathrm{mV}$	-

Table 3.2 Overview of the investigated initial membrane potentials with mean μ and standard deviation σ for normal distributions and a range for uniform distributions and cut off normal distributions. All normal distributions have been tested with a distribution that is cut off at the threshold value $V_{\mathrm{thr}} = 20\,\mathrm{mV}$ as well as one that is open.

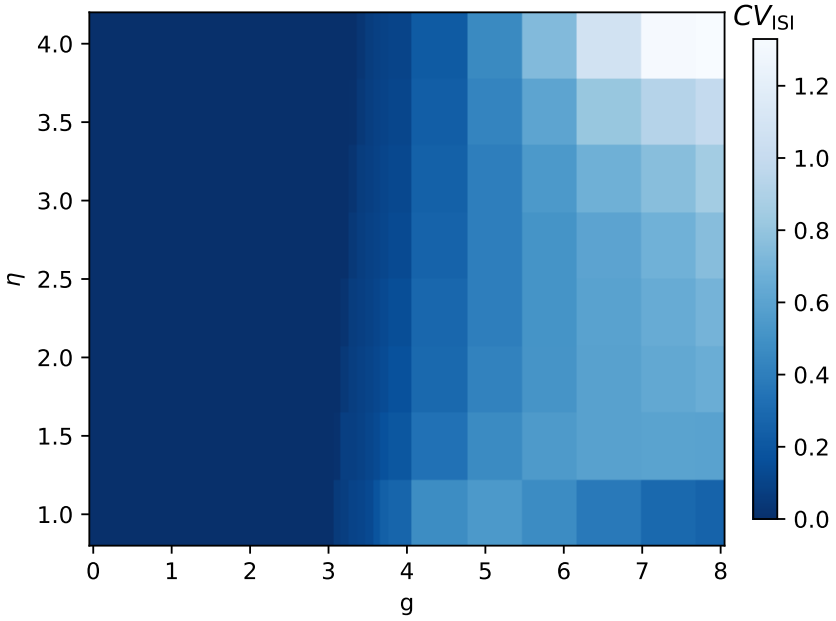
For all these initial membrane potential distributions the network behaviour stays the same and all the characteristics are approximately constant. Only in the SR regime some effect can be seen when using a distribution where about half of the neurons are initialised with a membrane potential that is higher than the threshold $v_{\mathrm{thr}} = 20\,\mathrm{mV}$. With this approach, networks in the SR state can be influenced to reliably build two neuron clusters, opposed to sometimes randomly building three or more (see figure 3.2a). This can examplatory be seen in figure 3.6a.

3.2.2 Distributed Neuron Parameters

This section investigates how the network reacts, when a variation is added to individual neuron parameters as well as to a combination of them.
When adding the variations in table 3.3 to the neuron parameters individually, none of these distortions do affect the states of the network enough to make it change its behaviour, except when looking at the delay D. This is shown in figure 3.7.
While there are some numerical differences in the characteristics of the neuron behaviour and global behaviour, synchronicity and regularity is kept in all cases, except when varying the delay D. With this parameter the network behaviour in the SR state clearly changes and the global behaviour becomes asynchronous. It is also the only case where the error is visible in figure 3.7, meaning that for different seeds the CV_{G} value is varying more, making the network unstable for a distributed delay D. Since this is the only parameter, that causes the networks behaviour to change significantly, it is investigated with more details in section 3.2.3.
Despite the fact that the other parameters do not lead to changes in the overall network behaviour, this does not necessarily hold when combining the variation for all these parameters. The effect on a network where every parameter is varied except the delay is depicted in figure 3.8 and explicitly listed in the last column of figure 3.7. As it can be



(a) Plot of the characteristic of the global activity CV_G.



(b) Plot of the characteristic of the neuron activity CV_{ISI}.

Figure 3.6 Exemplary plot for the CUBA LIF network with delay $D = 1.5\,\mathrm{ms}$ and a distribution of the initial membrane potentials. In this case a normal distribution with the range $-30\,\mathrm{mV}$ to $50\,\mathrm{mV}$ was used.

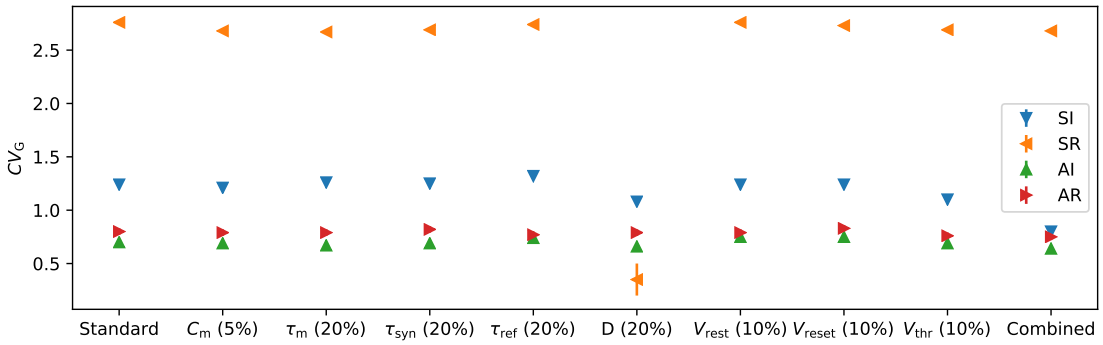
Parameter	Value	Standard deviation
C_{m}	1.0 pF	5 %
τ_{m}	20.0 ms	20 %
τ_{syn}	0.01 ms	20 %
τ_{ref}	2.0 ms	20 %
D	1.5 ms to 3.0 ms	20 %
V_{rest}	0.0 mV	10 %
V_{reset}	10.0 mV	10 %
V_{thr}	20.0 mV	10 %
ω_{exc}	0.1 nA	10 %
ω_{inh}	0.0 nA to 0.8 nA	10 %

Table 3.3 Overview of the standard deviation of the normal distribution each parameter is tested with.

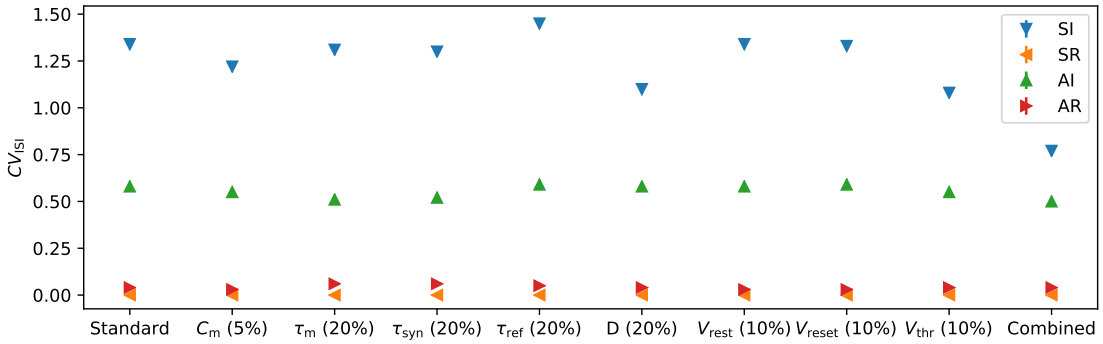
seen, this combination has the effect that the transition line between the SR regime for values with $g < 3$ and the AI regime for $g > 3$ does not go up diagonally anymore, but now the transition happens for all η values at around the same values of g. The transition line is also not as clear as it is without distributed parameters, but it smears out a bit into both regimes. This change in the transition line can be similarly found when looking at the regularity of the network.
Another effect this combination of varied parameters has is, that for SI states the CV_{G} value drops so far that it can not really be distinguished from the asynchronous states (see the last column of figure 3.7a). While there is still some increase in the CV_{G} value (see figure 3.7b) when looking at the areas where the maximal values reached are and can barely be distinguished from asynchronous behaviour (see figure 3.7). A third mentionable effect for this investigation is, that for small η values it appears to be more likely for the SI regime to be built of three clusters instead of two. All of these effects can probably be explained by the fact that the neuron parameters are distributed and therefore the neurons are behaving in the slightly different ways, leading to less correlation and therefore less synchronous behaviour (see section 2.3.2) in the network.

3.2.3 Distributed Delay

As mentioned in the previous section, distributing the delay D of a network is a special case when varying individual parameters, since it is the only parameter variation that leads to an actual change in behaviour on its own. To have a better understanding of this effect, in figure 3.9 it is plotted how the different network states react to an increasing standard deviation of the delay D. As it can be seen, the regularity of a network remains largely unaffected except for a small increase in CV_{ISI} for AR states from $CV_{\mathrm{ISI}} = 0.04$ to $CV_{\mathrm{ISI}} = 0.06$ and a slight decrease of the regularity in the SI state form $CV_{\mathrm{ISI}}1.38$ to

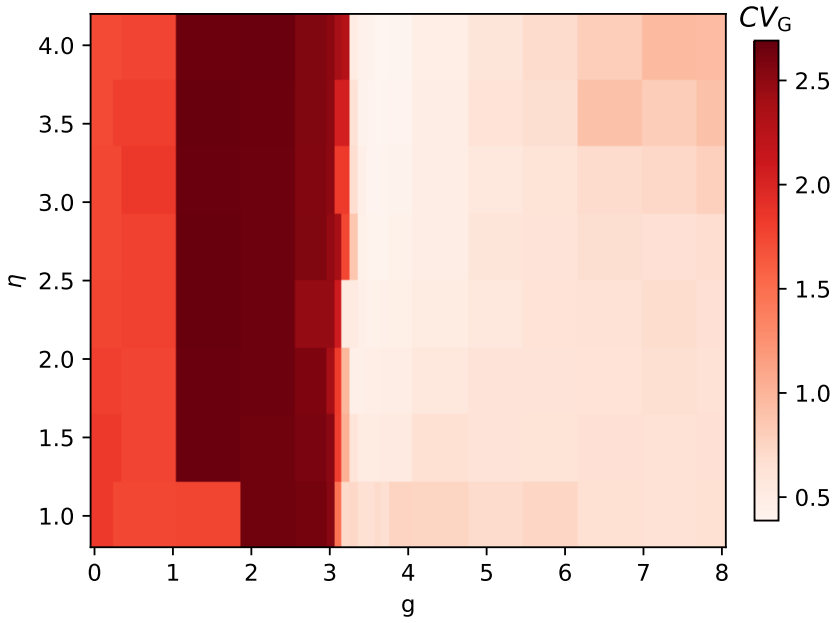


(a) Plot of the effect distributed parameters have on the global behaviour CV_G for each network state.

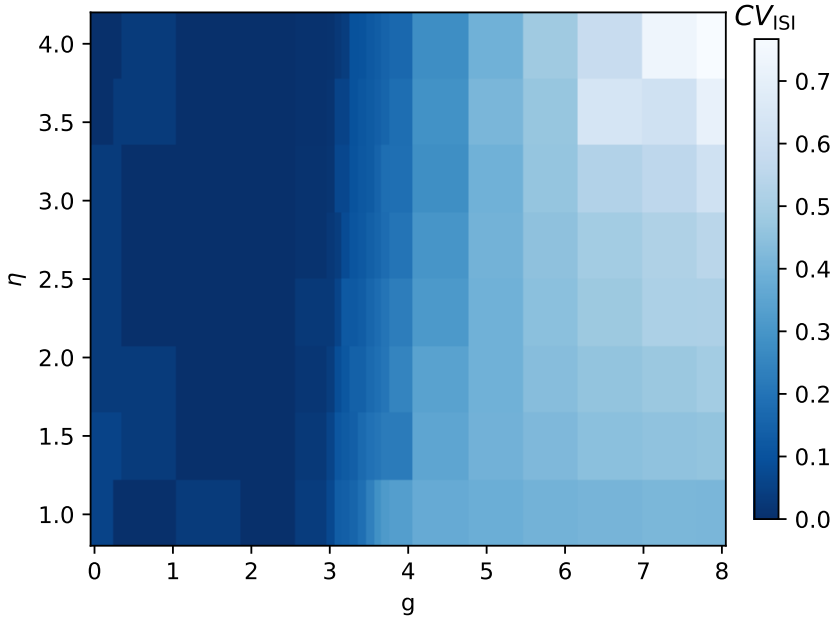


(b) Plot of the effect distributed parameters have on the neuron behaviour CV_{ISI} for each network state.

Figure 3.7 Depiction of how the CUBA LIF neuron network reacts to distributed neuron parameters. For each network state, the effect of adding a normal distribution to each of the neuron parameters is investigated individually. The standard deviation for each parameter is the value in brackets behind the parameter. The network parameters for the used different network states are: SI ($D = 1.5\,\text{ms}$, $g = 8.0$, $\eta = 4.0$), SR ($D = 1.5\,\text{ms}$, $g = 2.2$, $\eta = 2.2$), AI ($D = 1.5\,\text{ms}$, $g = 6.5$, $\eta = 1.9$), AR ($D = 2.0\,\text{ms}$, $g = 1.0$, $\eta = 3.5$). Additionally in the first column the characteristics for networks without distributed parameters is shown for reference and in the last column the characteristics of a network where all parameters are varied according to table 3.3, except for the delay. The values shown in the plot have been calculated by running multiple simulations with different seeds and calculating their mean and standard deviation, but the error bars are in most cases to small to be visible.



(a) Plot of the characteristic of the global activity CV_G.



(b) Plot of the characteristic of the neuron activity CV_{ISI}.

Figure 3.8 Plot of a CUBA LIF neuron network behaviour with delay $D = 1.5\,\mathrm{ms}$ in which all neuron parameters are varied according to 3.3 except for the delay D.

$CV_{\mathrm{ISI}} = 1.24$. In both cases this does not effect the behaviour too much. For the AI and SR state, the regularity stays approximately constant.
Regarding the change of synchronicity, the measured effects are more significant (see figure 3.9a). While for the SI and AI state, the synchronicity is unaffected by distributing the delay D, for both the SI and AR state the CV_{G} value decreases significantly. While for AI states this means that the network behaviour stays asynchronous, the SI state loses its synchronous behaviour and becomes asynchronous. The CV_{G} value of the SR state drops from around $CV_{\mathrm{G}} = 2.8$ to approximately $CV_{\mathrm{G}} = 0.7$. It is also apparent, that the only case in which error bars are visible in figure 3.9a are for the SI state, especially for distributions $0.1 < \sigma < 0.16$. In this range the SI state is unstable in the way that with different seeds the network randomly ends up in a synchronous behaviour or in a asynchronous.
When considering the synchronous regular (SR) state, (compare to 3.3a) one has several clusters of neurons that are all firing simultaneously. Almost all neurons are part of one of these clusters and the time between two clusters spiking is determined by the synaptic delay. In case of two clusters for example the spikes emitted by one cluster will cause the other cluster to spike and vice versa. This means that $3\,\mathrm{ms}$ pass between two spikes of a cluster, a time that is necessarily bigger than the refractory time of the neurons. This means, that the SR state is dependent on the synaptic delay, which suggests that it is sensitive to variations of this parameter. The network is therefore expected to lose the synchrony of the SI state, when the delay D of the Network is not fix anymore.

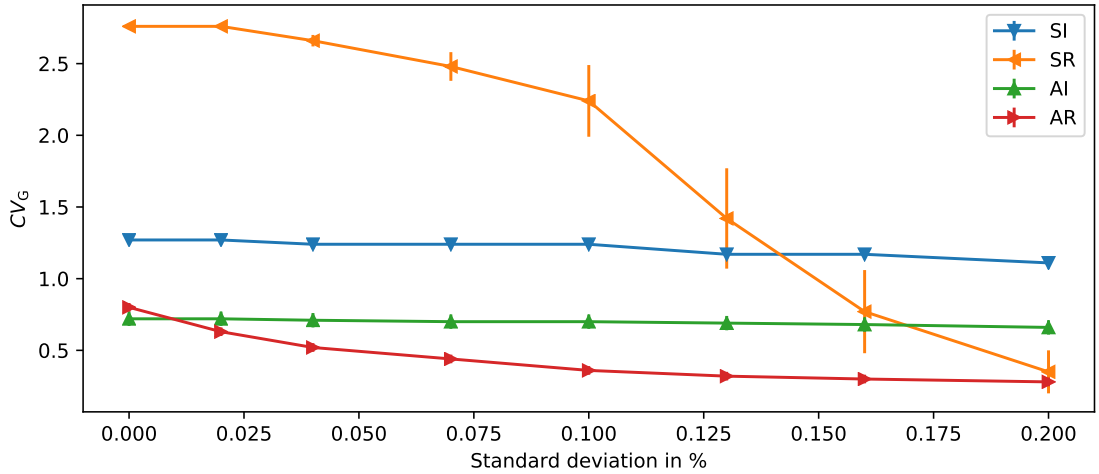
3.3 COBA LIF Neuron Network

To make another step towards the BrainScaleS system, the neuron model is changed to COBA LIF neurons, since it is supported by the hardware, while CUBA LIF neurons are not. To be more precise, the hardware can run simulations of COBA LIF neurons with an exponentially decaying kernel:

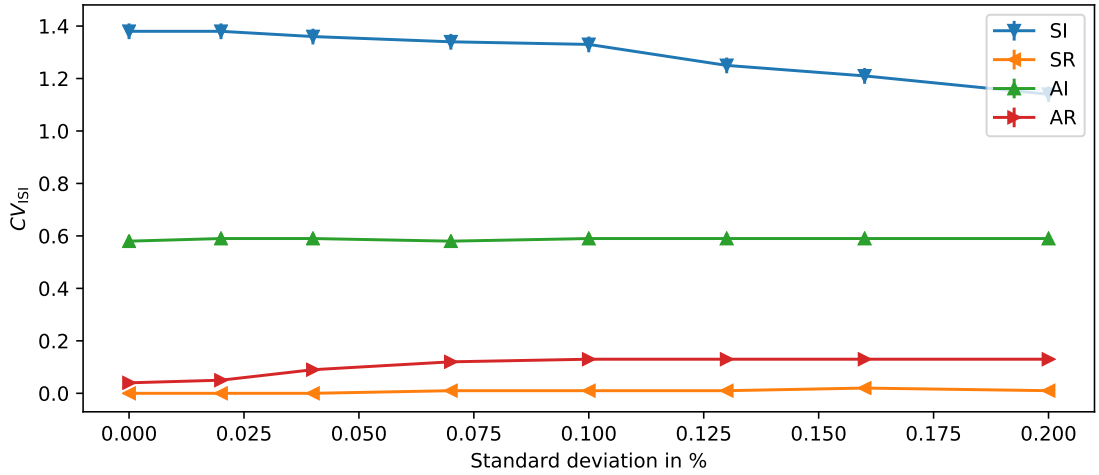
$$\varepsilon_n(t) = \Theta(t) \exp\left(-\frac{t}{\tau_{syn}}\right). \tag{3.5}$$

3.3.1 Transfer of the Network

For the implementation of the network with COBA LIF neurons, excitatory and inhibitory potentials have to be set. For the simulation of the network, only relative differences in the potential are of interest, since the values can be shifted as needed without having any effect on the network itself. While this does not hold up for hardware simulations, due to a limited range of potentials that can be set on the hardware, it was therefore focused on these relative potential differences. This choice was oriented on examples from the BrainScaleS guidebook (Andrew P. Davison et al. 2019), where a difference between excitatory

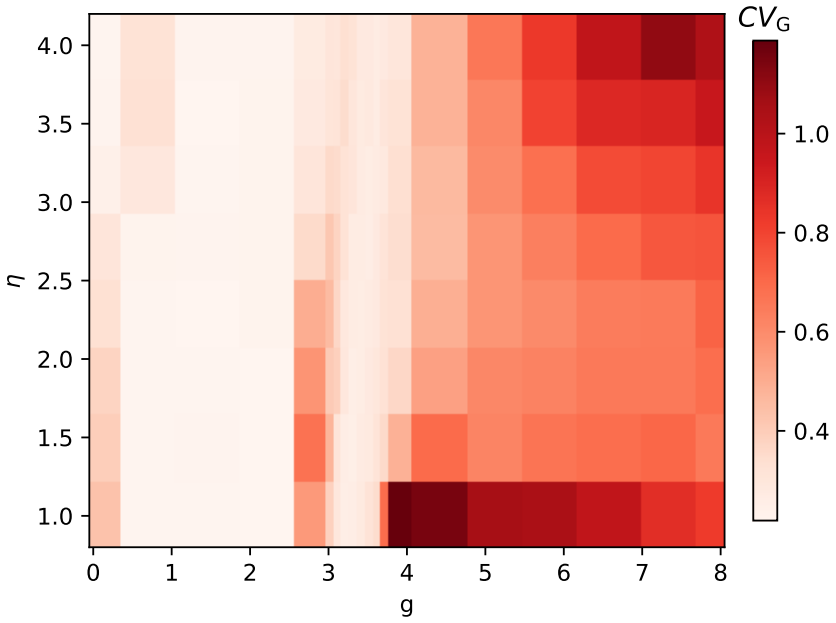


(a) Plot of the characteristic of the global activity CV_{G}.

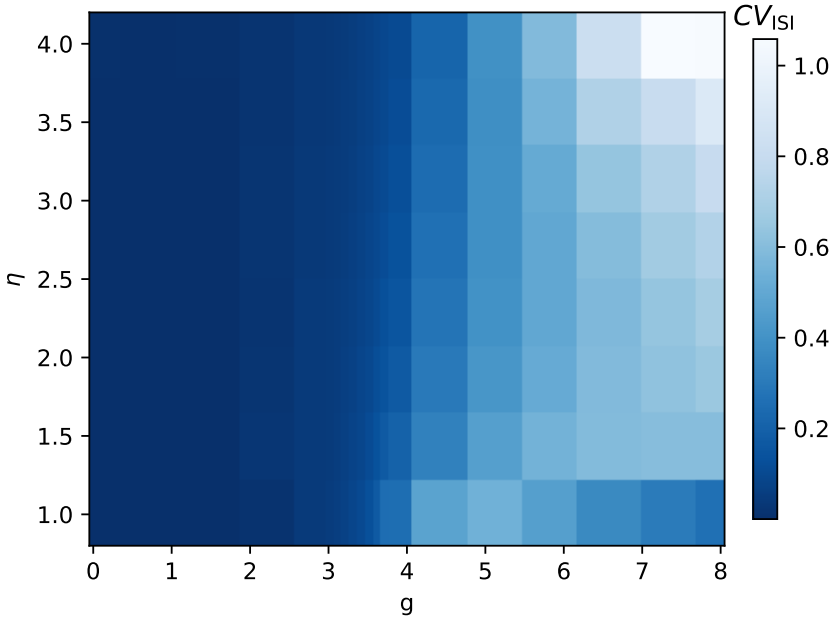


(b) Plot of the characteristic of the neuron activity CV_{ISI}.

Figure 3.9 Change in the CUBA LIF neuron network states in as function of increasingly wider distributed delay D. The values shown in the plot have been calculated by running multiple simulations with different seeds and calculating their mean and standard deviation, but the error bars are in most cases to small to be visible.



(a) Plot of the characteristic of the global activity CV_G.



(b) Plot of the characteristic of the neuron activity CV_{ISI}.

Figure 3.10 Plot of a CUBA LIF neuron network behaviour with a distributed delay D with standard deviation of 20 %.

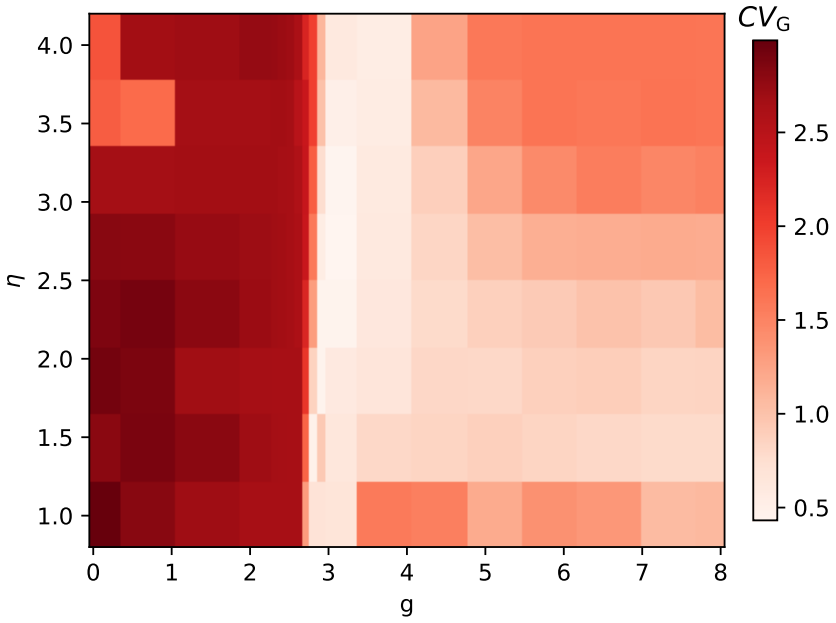
Symbol	Value	Description
C_m	$1.0\,\mathrm{pF}$	Capacitance of the neuron
τ_{syn}	$0.01\,\mathrm{ms}$	Synaptic time constant
V_{exc}	$90\,\mathrm{mV}$	Excitatory potential
V_{inh}	$-70\,\mathrm{mV}$	Inihibitory potential
g_L	$5.0\,\mathrm{pS}$	Conductance to the leak potentials
ω	$0.000\,13\,\mu\mathrm{S}$	Calculated synaptic weight
dt	$0.1\,\mathrm{ms}$	Simulation time step
T	$2000\,\mathrm{ms}$	Duration of the simulation

Table 3.4 Additional parameters used for the simulation of the network with COBA LIF neurons. The network parameters can be found in 2.1.

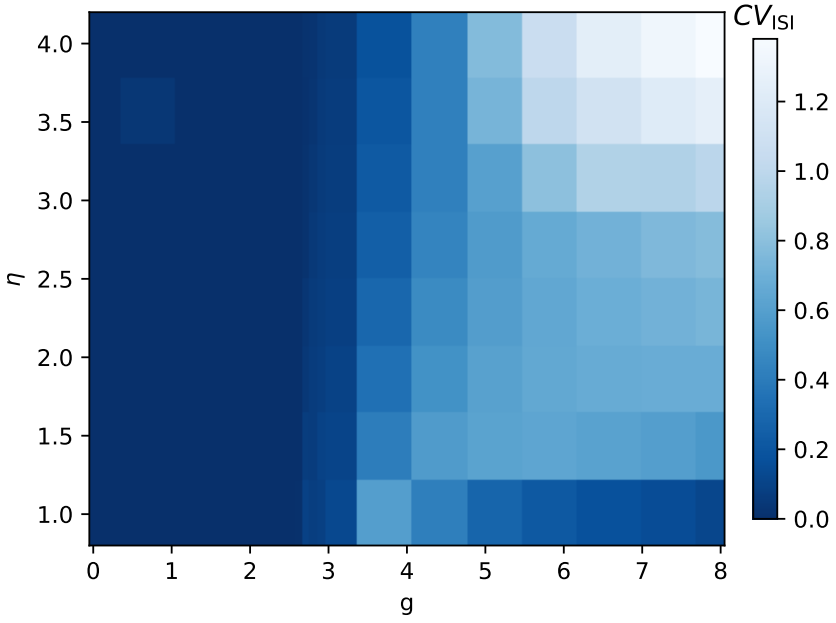
and inhibitory potential of $160\,\mathrm{mV}$ was used.
With this range in mind, the excitatory potential is set to $V_{\mathrm{exc}} = 90\,\mathrm{mV}$ and the inhibitory potential is set to $V_{\mathrm{inh}} = -70\,\mathrm{mV}$. This means that the reset potential $V_{\mathrm{reset}} = 10\,\mathrm{mV}$ is in the middle of the two potentials. Similarly to the implementation of the network with CUBA neurons (section 3.1), the synaptic weight for this setup is set in a way that the membrane potential of a neuron, whose membrane potential is centred between the excitatory and the inhibitory potential at $u = 10\,\mathrm{mV}$, changes its value according to the network architecture by $\Delta u = 0.1\,\mathrm{mV}$. The synaptic weight is experimentally deduced to be $\omega = 0.000\,13\,\mu\mathrm{S}$, by simulating only one excitatory spike arriving at a neuron and reading ot the change in maximal change in membrane potential. The setup of the network is set in this way, because after each spike, when the neurons are reset to their reset potential $V_{\mathrm{reset}} = 10\,\mathrm{mV}$, the change in membrane potential is calibrated to this potential. The simulation of a network according to this setup, the parameters are summed up in table 3.4, leads to results that are compatible with the results of the CUBA LIF neuron network without any parameter distributions. The results can be found in figure 3.11. When comparing this figure to figure 3.2, the CUBA LIF neuron network without any parameter distributions, the characteristics of the SR, SI, and AI regimes in the network stay approximately the same. There is however a shift of these regimes happening. The transition line from the SR regime to the AI regime now happens at $2.7 < g < 3.1$. The SI regime in the upper right corner starts at approximately $g > 4.5$ and $\eta > 2.8$, the SI regime for $\eta = 1$ is at $3.5 < g < 4.8$. Possible reasons for these shifts are discussed in the next section.

3.3.2 Possible Influences on the Network

As it is explained in section 2.1, the transition from CUBA to COBA neurons is more complicated than from δ-input based neurons to CUBA neurons. This suggests to do some theoretical calculations, especially regarding the effective membrane time (equation 2.8) and how one can expect it to differ from the constant membrane time. The approach



(a) Plot of the characteristic of the global activity CV_G.



(b) Plot of the characteristic of the neuron activity CV_{ISI}.

Figure 3.11 Plot of the characteristics of the network with COBA LIF neurons. In the area between $2.6 \leq g \leq 3.2$ more data points have been recorded than for the other parts of the plot, since this allows the depiction of the transition line between the different network states more easily and recreating this transition line is one of the more interesting parts of the plot.

Symbol	Value
g_L	5.0 pS
$g_{\text{exc, ext}}$	0.15 pS to 0.6 pS
$g_{\text{exc, int}}$	0.15 pS to 6.0 pS
$g_{\text{inh, int}}$	0.04 pS to 2 pS

Table 3.5 Comparison of the different occurring mean conductances in conductance based neurons. g_L is the mean conductivity of the LIF neuron, $g_{\text{exc, ext}}$ is the mean conductivity caused by the external inputs, $g_{\text{exc, int}}$ is the mean conductivity caused by excitatory internal intputs, $g_{\text{inh, int}}$ is the mean conductivity caused by inhibitory internal intputs.

taken for this consideration is to calculate the mean effective membrane time. To do, so equation 2.8 is rearranged to the following form:

$$\overline{\tau_{\text{eff}}} = \frac{C_m}{\overline{g_L} + \overline{g_{\text{exc, ext}}} + \overline{g_{\text{exc, int}}} + \overline{g_{\text{inh, int}}}}, \tag{3.6}$$

where a distinction between spikes coming from neurons in the network (int) and from external sources (ext) was added. This was done because, while the spikes from external sources follow a Poisson distribution, which makes the mean a valid measurement for this quantity, the spikes in between neurons are, for synchronous global behaviour, expected to be correlated. This means that at some times many spikes arrive at once at a neuron, while at other times only few spikes other than the spikes from the external sources arrive at the neuron. This could lead to a lot of spikes at one time and none at another point in time, in total making the mean not a meaningful quantity. Since the purpose of this calculation is only to estimate the effective membrane time, this should be good enough. The mean of the leak resistor is given by $\overline{g_L} = g_L$, the mean of the other terms g_n is given by the following equation:

$$\overline{g_n} = \lim_{T\to\infty} \int_0^T \frac{1}{T} \sum_s \omega_n \exp\left(-\frac{t}{\tau_{\text{syn}}}\right) \mathrm{d}t \tag{3.7}$$

$$= \nu_n \omega_n \tau_{syn}, \tag{3.8}$$

where ν is the mean frequency the corresponding spikes are arriving at the neuron. For this network configuration, the individual neurons are spiking with a mean frequency in the range of $400\,\text{Hz} > \nu_{\text{neuron}} > 10\,\text{Hz}$, where higher frequencies are obtained for small g values and with an increase in g the frequency is decreasing. Together with the number of connections of a neuron and with formula 2.9, the rate of the external input, this leads approximately to the conductances given in table 3.5. As mentioned earlier, only the external input is reliably following a Possonian distribution, while spikes in between the neurons might be correlated. This is especially true for the SR regime, i. e. where $g < 3.5$, where the highest neuron spiking frequencies are reached. While these estimations may

Membrane potential u	Δu_{exc}	Δu_{inh}
0 mV	0.12 mV	0.09 mV
5 mV	0.11 mV	0.10 mV
10 mV	0.10 mV	0.10 mV
15 mV	0.10 mV	0.11 mV
20 mV	0.09 mV	0.12 mV

Table 3.6 An overview of how the change in membrane potential changes with different membrane potentials when using COBA neurons. The second row (Δu_{exc}) lists the potential changes for excitatory spikes, the third one (Δu_{exc}) for inhibitory spikes.

therefore not lead to the most reliable results, it follows that the effective membrane time constant is to some degree smaller than the membrane time constant of the original network architecture. Since a smaller membrane time constant leads to a faster decay of the membrane potential u to its resting potential V_{rest}, this effect can be compared to the effect higher inhibitory weight would have on the network and thereby shifting the network regions to smaller g values, which is exactly what is seen in figure 3.11.
Another aspect that should be taken into consideration is how the change in membrane potential is affected by the current membrane potential. For this, the change in membrane potential after a spike has been measured for several membrane potential values between $u = 0\,\text{mV}$ and $u = 20\,\text{mV}$, the results are listed in table 3.6. What this means is, that if the excitatory and inhibitory potentials are not centred around the mean membrane potential of a neuron, either inhibitory or excitatory spikes, whichever potential is has the bigger difference to the mean membrane potential, will have effectively more weight than the other spikes. This would therefore also lead to a shift along the g axis.

3.4 COBA LIF Network With Distributed Parameters

This section takes a look at what happens when the effects discovered in the previous sections are combined. This means that all parameters will be distributed according to table 3.3, including the delay D and that the network is based on COBA LIF neurons. Within this setup, one expects that all synchronous behaviour of the network disappears, as the synchronicity of the SR state decreases with the distributed delays D (see section 3.2.3) and the synchronicity of the SI state decreases with the distribution of all the other parameters (see section 3.2.2).
This is exactly what is seen with this combination. As depicted in figure 3.12, the characteristic for the global behaviour is in no case higher than $CV_{\text{G}} = 0.55$, which is well in the asynchronous behaviour as it became clear previously.
While this means that the regimes that behave synchronous in the original design have

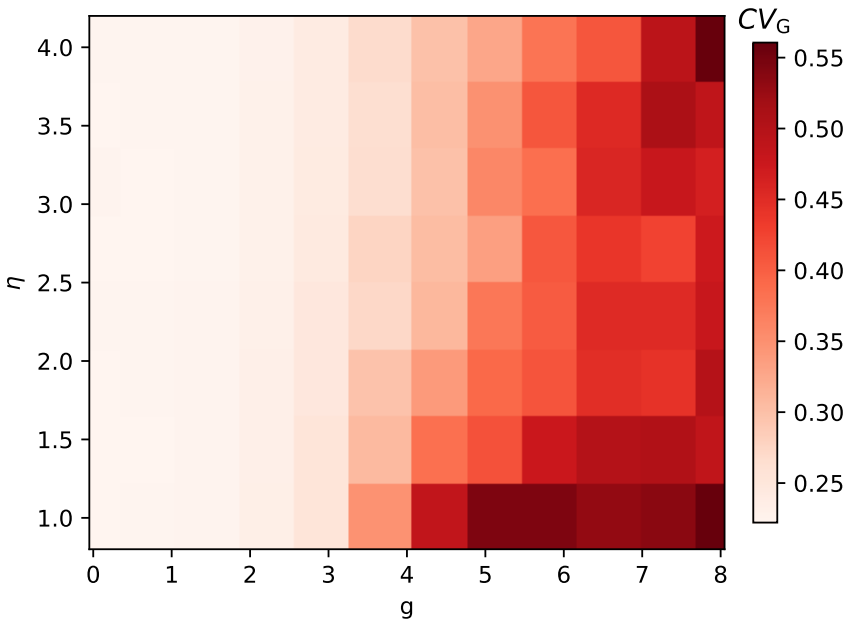
Synaptic time constant τ_{syn}	**Synaptic Weight ω**
$0.01\,\mathrm{ms}$	$10.0\,\mathrm{pA}$
$0.1\,\mathrm{ms}$	$1.03\,\mathrm{pA}$
$0.5\,\mathrm{ms}$	$0.220\,\mathrm{pA}$
$1.0\,\mathrm{ms}$	$0.118\,\mathrm{pA}$
$1.5\,\mathrm{ms}$	$0.0823\,\mathrm{pA}$
$2.0\,\mathrm{ms}$	$0.0660\,\mathrm{pA}$

Table 3.7 Summary of adjusted weights for different synaptic time constants with current based neurons. The resulting total change in membrane potential is $0.1\,\mathrm{mV}$ in each case. Each parameter was determined to three significant digits.

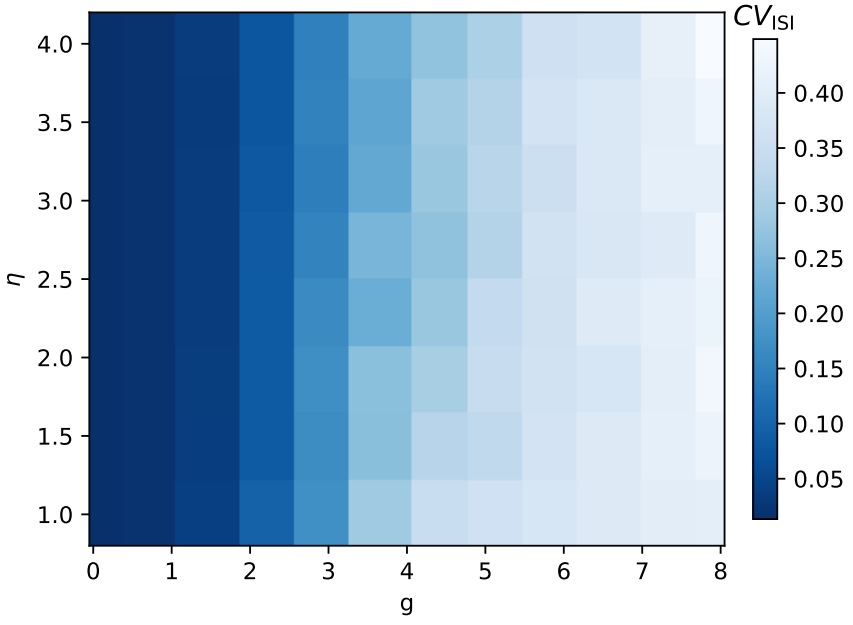
changed their behaviour a lot, looking at the asynchronous regimes, their behaviour did not only stay unchanged, but the CV_G value for the AI regime is only slightly lower ($\Delta CV_\mathrm{G} = 0.1$) than it is discovered when looking at the standart CUBA LIF neuron network. The asynchronous behaviour of the network is therefore quite stable with respect to all changes applied to the network.
Looking at the regularity of the network, it is apparent that the increase in regularity does not happen as rapid as in previous cases and that the initial ascend starts earlier. The latter point is partly due to the transition line being shifted to smaller g values, as seen in section 2.1.3. It also does not increase for high g and η values that can be seen in figure 3.2 but seems to be independent of η, which is expected to some degree as this effect was already discovered in section 3.2.3. For the regularity this means that the regular states of the network are widely unaffected by the changes made to the network, while for the irregular states the CV_ISI is in general smaller than for networks without distributed parameters, but not small enough to be seen as regular. This means that for both cases, while the characteristics of the regularity may change a bit the overall behaviour stays the same.

3.5 Prolonged Synaptic Time

The synaptic time constant set for the initial reproduction $\tau_\mathrm{syn} = 0.01\,\mathrm{ms}$ was explicitly chosen to resemble the original network, but is much smaller than the adjustable range on BrainScaleS. On BrainSaleS synaptic time constants can not be much smaller than $2.0\,\mathrm{ms}$ (S. Schmitt: private communication). Thus another necessary test on the network is the change in behaviour when choosing larger synaptic time constants.
The weight ω was adjusted for every synaptic time constant in a way, that the total change in membrane potential still matched the wanted potential difference. This was done experimentally, by simulating only one excitatory spike arriving at a neuron and reading ot the change in maximal change in membrane potential, since the accuracy of equation 2.2



(a) Plot of the characteristic of the global activity CV_G.



(b) Plot of the characteristic of the neuron activity CV_{ISI}.

Figure 3.12 Network with COBA LIF neurons and all parameters are distributed according to table 3.3.

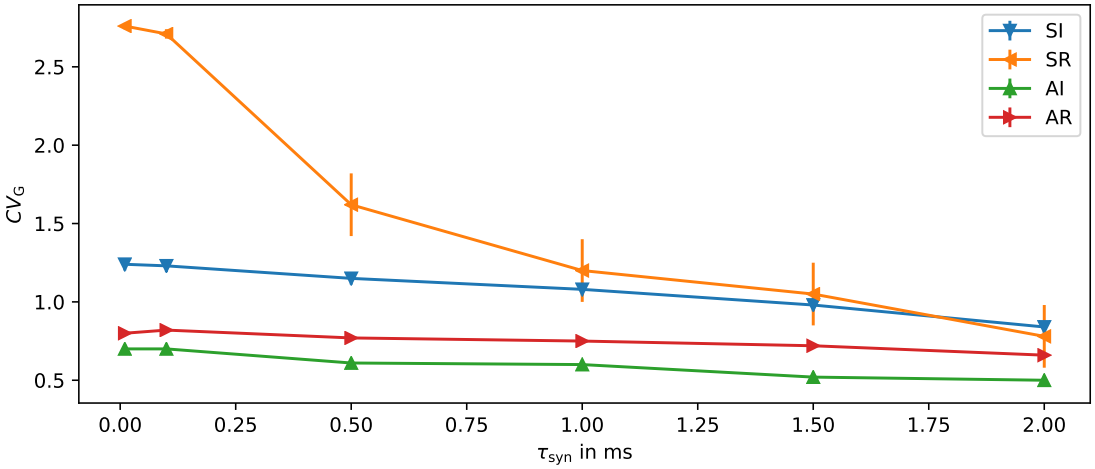
decreases with greater synaptic time constants because the assumption $\tau_{\mathrm{syn}} = 2.0\,\mathrm{ms} \ll \tau_{\mathrm{m}} = 20\,\mathrm{ms}$ is not true anymore. Tested synaptic time constants along with the adjusted weight can be looked up in table 3.7.

Without changing any other parameters in the network than the synaptic time constant τ_{syn} and respectivly the synaptic weight ω else in the network, this leads to a decrease of the CV_{G} value for every network state (see figure 3.13a). But while this decrease is relatively small for asynchronous states, about $\Delta CV_{\mathrm{G}} = 0.2$ for AI states and $\Delta CV_{\mathrm{G}} = 0.15$, and also not leading to a change in the overall behaviour since they already are behaving asynchronous, the effect on synchronous states is bigger. For SI states the longer synaptic time leads to a drop of about $\Delta CV_{\mathrm{G}} = 0.4$ and for SR states of approximately $\Delta CV_{\mathrm{G}} = 2.0$. As shown in figure 3.13a this means that the previously synchronous behaving states are barely distinguishable from the asynchronous states when prolonging the synaptic time constant to $\tau_{\mathrm{syn}} = 2.0\,\mathrm{ms}$.

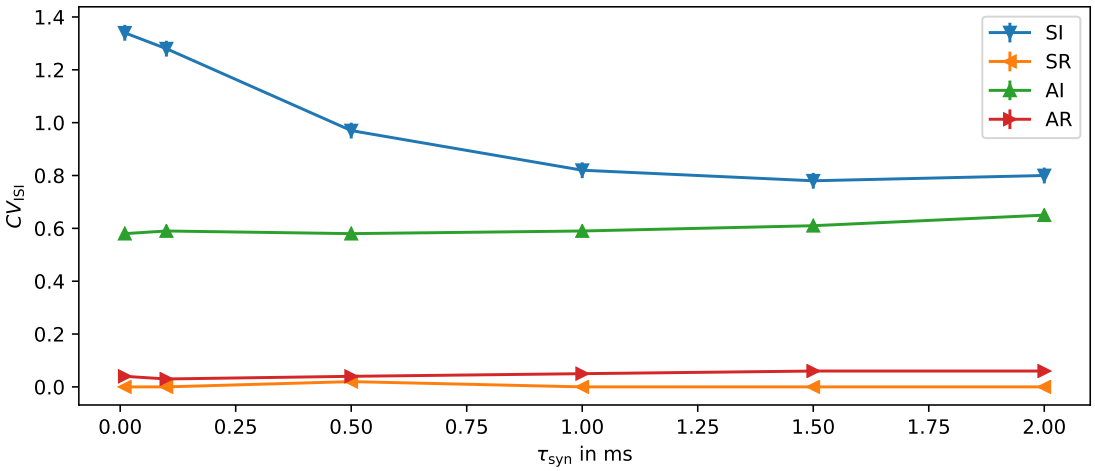
Regarding the neuron behaviour, the regularity stays approximately constant for SR, AI and AR states, and drops about $\Delta CV_{\mathrm{ISI}} = 0.5$ for SI states (see figure 3.13b), which is however not enough to change the behaviour of the network from irregular to regular.

The behaviour of the network with a synaptic time constant of $\tau_{\mathrm{syn}} = 2.0\,\mathrm{ms}$ is depicted in figure 3.14. What can be seen when looking at the global behaviour at the area $g < 3.5$, where for the standard network the SR regime has been, one can see that the CV_{G} value is distributed in the range $0.7 < CV_{\mathrm{G}} < 1.6$, meaning that the global behaviour is unstable in the area with sometimes behaving on a more synchronous way and most of the times in a asynchronous one. This also explains the comparably large error bars depicted in figure 3.13a for SR states.

The plot for the neuron behaviour (figure 3.14b) looks as expected, as it is only missing the increase of the CV_{ISI} value, which is already discussed above.

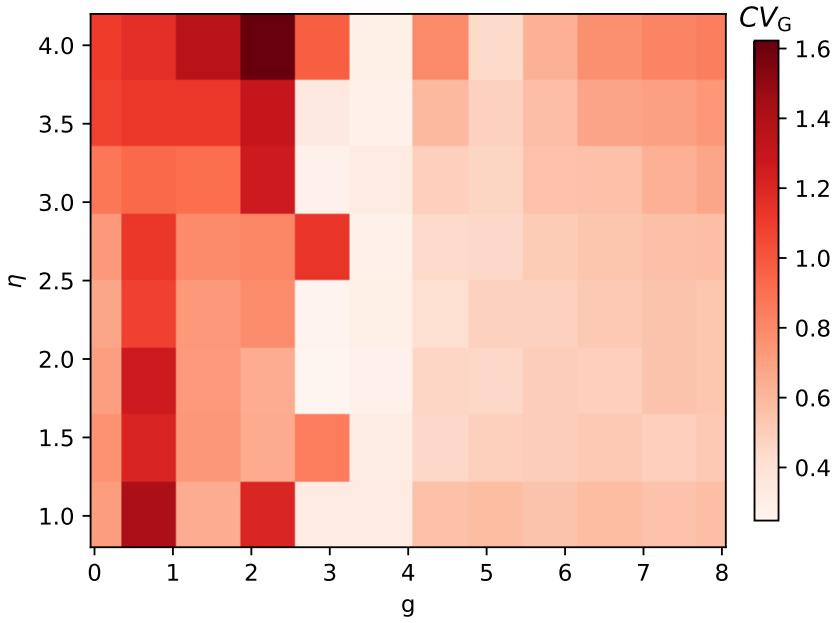


(a) Plot of the characteristic of the global activity CV_G.

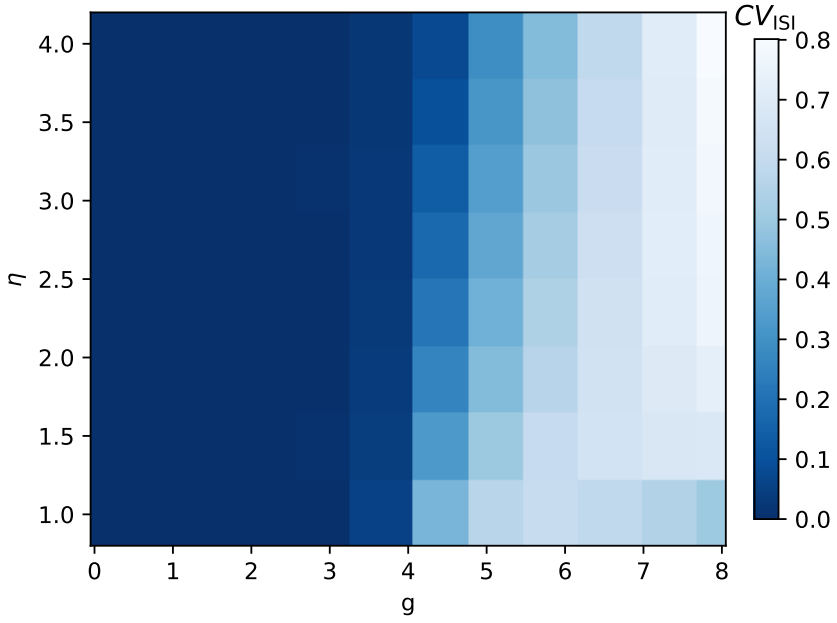


(b) Plot of the characteristic of the neuron activity CV_{ISI}.

Figure 3.13 Depiction of how the network reacts to prolonged synaptic time constants. For each network state, the effect of prolonging the synaptic time constant is investigated. The network parameters for each of the used networks are: SI ($D = 1.5\,\text{ms}$, $g = 8.0$, $\eta = 4.0$), SR ($D = 1.5\,\text{ms}$, $g = 2.2$, $\eta = 2.2$), AI ($D = 1.5\,\text{ms}$, $g = 6.5$, $\eta = 1.9$), AR ($D = 2.0\,\text{ms}$, $g = 1.0$, $\eta = 3.5$). The values shown in the plot have been calculated by running multiple simulations with different seeds and calculating their mean and standard deviation.



(a) Plot of the characteristic of the global activity CV_G.



(b) Plot of the characteristic of the neuron activity CV_{ISI}.

Figure 3.14 Plot of a CUBA LIF neuron network behaviour with delay $D = 1.5\,\mathrm{ms}$ and a synaptic time constant of $\tau_{syn} = 2.0\,\mathrm{ms}$.

Chapter 4

Discussion

With regards to the ongoing implementation of the Cortical Column Network (Potjans and Diesmann 2012) on BrainScaleS I, this thesis investigates the effects the transition of a network from software simulation to emulation on hardware are causing. This is done because, due to imperfections in the manufacturing process, parameters on an analogue chip are never going to be completely as expected and even with calibrations one will not be able reach perfect results. For this thesis, these effects have been estimated with a Gaussian variance, where the standard deviation for temporal parameters is set to $20\,\%$ of the parameter value, for voltage parameters to $10\,\%$ and for capacities to $5\,\%$. Other aspects that are considered are how the network reacts to LIF neurons with either current or conductance based synaptic models and how a change in the synaptic time constant affects the network. This investigations are performed with the network described in Brunel (2000), a network that shows close resemblance to some parts of the Cortical Column Network.

The first part of the thesis 3.1 concentrates on reproducing and verifying the results of Brunel (2000) with software simulations. This is done successfully in section 3.1.1, all network regimes are found where they are expected. Being able to reproduce existing results has the advantage that possible mistakes in the implementation of the network are minimised and the network works as it should do. On the other hand this reproduction produces a baseline of characteristics of the network which help comparing further results with the network. Having some idea what to expect also helps with verifying that the analysis methods are well suited to describe the networks behaviour, as discussed in section 3.1.2.
The next part (section 3.2) investigates how the individual parameters of the network react to variations and how different combinations of these variations influence the network. Regarding differently distributed initial membrane potentials (see section 3.2.1), the network behaves very stable with regards to all variations. While only looking at the individual parameter variations, only the synaptic delay D had enough influence to change a networks behaviour significantly, i. e. change its state. As it can be seen in section 3.2.3, a distributed delay has a significant influence on the SR states of the network, since almost all synchronous behaviour is lost. Varying all the remaining parameters at once (see section 3.2.2) also lead to changes in the behaviour of the network. This time the SI state was affected most significantly, while the other states keept their behaviour. The SI state lost its synchronous behaviour in this case. Summarizing the change in global behaviour, the variation of the parameters mosly affected the synchronous parts of the network. The characteristics of the asynchronous parts of the networks have stayed ap-

proximately constant, except when varying the delay D, where in the AR state the CV_{G} value decreased with higher standard deviations. During the implementation of the Cortical Column Network (Potjans and Diesmann 2012), which is behaving asynchronous, on BrainScaleS I this means that to keep the characteristics of the network constant, some attention should be given to how the delay D is handled.
The transition of the neuron model from CUBA LIF neurons to COBA LIF neurons is investigated in section 3.3. While this transition does not change the network behaviour of the different regimes of the network, the extent of the regimes changes and they are also slightly shifted with the used parameters for the COBA LIF network. It may be possible to counteract these effects to some extent, for example by choosing different excitatory and inhibitory potentials for the COBA LIF neurons, but this has not been done as part of this thesis. Apart from eventual corrections of the parameter for the network with COBA LIF neurons, it has been shown that the transition to COBA LIF neurons works fine for the network. With this in mind the transition for the Cortical Column Network should work as well.
These individual results, i. e. having distributed neuron parameters and simulating with COBA LIF neurons instead of CUBA LIF neurons have been combined and tested all at once, with the expected result that especially the synchronous behaviour is sensitive to the variations, while asynchronous behaviour could be reproduced with similar characteristics at the expected places, once one takes the shift of the network regimes, due to the transition to COBA LIF neurons into account.
The final investigation on how a prolonged synaptic time constant influences the investigated network has been done separately, because this only affects networks, that have a time constant that is not in the range of time constants achievable on BrainScaleS I. For the investigated network that had naturally a very short synaptic time constant prolonging the time constants did once again cause the network to lose its synchronous behaviour, but also had measurable effects on the regularity. What has been left out for this case is to investigate how the behaviour is influenced when using different synaptic weights. This might be another interesting point to look at since this would mean that the membrane potential experiences a bigger change immediately after the neuron has received a spike.

Finally, I would like to talk about my working experience in the Electronic Vision(s) Group. I have to admit that I was very shy when I started working with this group and that I wanted to do anything on my own, without any help. This has lead to several problems during this thesis that could have been solved a lot faster, or even avoided all together if I had just asked for help. Once I allowed myself to go to other people for help, even if the question might be stupid, it turned out that everyone was willing to help and would not be mad if the answer to my question was obvious. During my time with the group I learned a lot about working as part of a big group, with many people specialising in several very different topics, for example how important communication is for a successful project.

Chapter 5

Outlook

Sadly I have not been able to implement the network on hardware and actually test the predictions from this thesis. Doing this could help by either confirming the results about the change in behaviour of the network that have been made in this thesis or give clues about which assumptions in this thesis are not valid.
One could also do some further research in the direction of changing the synaptic time constant of a network without changing the overall behaviour too much. A step in this direction has for example been done by Korcsak-Gorzo (2015), where a similar but smaller network with a larger synaptic time constant was investigated.
The investigations in this thesis have been done with continuous synaptic weights. This is fine for the scope of this thesis, but on BrainScaleS I this is not possible. The system only allows for discrete synaptic weights (4 bits plus offset). The compatibility with these discrete weights would not only pose an interesting research topic, but would also support the general goal of this thesis, to pave the way for the implementation of the Cortical Column Network (Potjans and Diesmann 2012), which also has to deal with the problem of modelling the network as originally described as good as possible with the available weights.
Another point that is mainly affecting the Cortical Column Network is the size of the network. To fit on one wafer, the cortical column has to be scaled down to approximately $5\,\%$ to $10\,\%$ of the original network (Weidner 2019).
There are several ways to scale down a network. One can for example just scale the neuron size and leave the connection probability, the synaptic weights, the external input and all the other parameter the same, but one can also adjust the synaptic weight and or the external input by some factors. Depending on how much the network has to be scaled down and which properties of the networks should be preserved (synchronicity, regularity), different approaches should be used. For many specific cases, approaches for scalability have been described in literature, for example one that might be interesting for the Cortical Column network: Sacha Jennifer van Albada, Helias, and Diesmann (2015).

References

Albada, Sacha J. van et al. (2018). "Performance Comparison of the Digital Neuromorphic Hardware SpiNNaker and the Neural Network Simulation Software NEST for a Full-Scale Cortical Microcircuit Model". In: *Frontiers in Neuroscience* 12, p. 291.

Albada, Sacha Jennifer van, Moritz Helias, and Markus Diesmann (2015). "Scalability of Asynchronous Networks Is Limited by One-to-One Mapping between Effective Connectivity and Correlations". In: *Computational Biology* 11.9. Ed. by Peter E. Latham.

Andrew P. Davison, Andrew et al. (2019). *HBP Neuromorphic Computing Platform Guidebook*. URL: https://electronicvisions.github.io/hbp-sp9-guidebook/index.html#.

Brandes, Ralf, Florian Lang, and Robert F. Schmidt (2019). *Physiologie des Menschen. mit Pathophysiologie*. ger. 32nd ed. Springer-Lehrbuch. Berlin, Heidelberg: Springer.

Brette, Romain and Wulfram Gerstner (2005). "Adaptive Exponential Integrate-And-Fire Model As An Effective Description Of Neuronal Activity". In: *Journal of neurophysiology* 94, pp. 3637–42.

Brunel, Nicolas (2000). "Dynamics of Sparsely Connected Networks of Excitatory and Inhibitory Spiking Neurons". In: *Journal of Computational Neuroscience* 8.3, pp. 183–208.

Brunel, Nicolas and Mark C. W. van Rossum (2007). "Lapicque's 1907 paper: from frogs to integrate-and-fire". In: *Biological Cybernetic* 97.5, pp. 337–339.

Davison, Andrew et al. (2009). "PyNN: a common interface for neuronal network simulators". In: *Frontiers in Neuroinformatics* 2, p. 11.

Furber, S. B. et al. (2014). "The SpiNNaker Project". In: *Proceedings of the IEEE* 102.5, pp. 652–665.

Gerstner, Wulfram, Werner M. Kistler, et al. (2014). *Neuronal Dynamics: From Single Neurons to Networks and Models of Cognition*. Cambridge University Press.

Gerstner, Wulfram, Andreas K. Kreiter, et al. (1997). "Neural codes: Firing rates and beyond". In: *Proceedings of the National Academy of Sciences* 94.24, pp. 12740–12741. eprint: https://www.pnas.org/content/94/24/12740.full.pdf.

Grimmett, Geoffrey and Dominic Welsh (2014). *Probability. An Introduction*. 2nd ed. Oxford University Press.

Hodgkin, A. L. and A. F. Huxley (1952). "A quantitative description of membrane current and its application to conduction and excitation in nerve". In: *The Journal of Physiology* 117.4, pp. 500–544. eprint: https://physoc.onlinelibrary.wiley.com/doi/pdf/10.1113/jphysiol.1952.sp004764.

Kleider, Mitja (2017). "Neuron Circuit Characterization in a Neuromorphic System". PhD thesis. University of Heidelberg.

Korcsak-Gorzo, Agnes (2015). "Firing States of Recurrent Leaky Integrate-and-Fire Networks". Bachelor Thesis. Heidelberg University.

Kruskal, Peter et al. (2007). "A binless correlation measure reduces the variability of memory reactivation estimates". In: *Statistics in medicine* 26, pp. 3997–4008.

Petrovici, Mihai, Johannes Bill, and Andreas Hartl (2017). *Brain-ispired Computing. Summer 2017*.

Peyser, Alexander et al. (2017). *NEST 2.14.0*.

Piccinini, Gualtiero and Sonya Bahar (2013). "Neural Computation and the Computational Theory of Cognition". In: *Cognitive Science* 37.3, pp. 453–488. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/cogs.12012.

Potjans, Tobias C. and Markus Diesmann (2012). "The Cell-Type Specific Cortical Microcircuit: Relating Structure and Activity in a Full-Scale Spiking Network Model". In: *Cerebral Cortex* 24.3, pp. 785–806. eprint: http://oup.prod.sis.lan/cercor/article-pdf/24/3/785/14099777/bhs358.pdf.

PyNN: documentation (2019).

Rudolph-Lilith, Michelle, Mathieu Dubois, and Alain Destexhe (2012). "Analytical Integrate-and-Fire Neuron Models with Conductance-Based Dynamics and Realistic Postsynaptic Potential Time Course for Event-Driven Simulation Strategies". In: *Neural Computation* 24.6, pp. 1426–1461.

Schemmel, Johannes et al. (2010). "A Wafer-Scale Neuromorphic Hardware System for Large-Scale Neural Modeling". In: *Proceedings of 2010 IEEE International Symposium on Circuits and Systems*.

Schmidt, Dominik (2014). "Automated Characterization of a Wafer-Scale Neuromorphic Hardware System". MA thesis. Kirchhoff-Institute for Physics.

Weidner, Jonas (2019). "Experiment Visualization and Simulations towards a Cortical Microcircuit on the BrainScaleS Neuromorphic Hardware". Bachelor Thesis. Heidelberg University.

List of Figures

List of Tables

Acknowledgements

I would like to thank:

- Dr. Johannes Schemmel for giving me the change to participate in his group and letting me work in the field of neuromorphic computing
- Sebastian Schmitt for providing me with a bachelor thesis, superivising, correcting, and helping me with the my thesis
- Hartmut Schmidt for always being ready to make some time when i need help and working with me on this thesis
- Christian Mauch, Jonas Weidner, Phillip Spilger and Josha Illmberger for answering all remaining questions and helping me with all problems I encountered
- The TMA people for helping me understanding my results
- All the other people in the Electronic Vision(s) Group, especially the ones working in the container building, for providing a beautiful, memorable and sometimes very emotional working atmosphere
- Mathieu Kaltschmidt and Thorben Frey for proofreading everything
- My family and friends for supporting me throughout my studies

Declaration of Authorship

I certify that this thesis is the product of my own work and no other than the cited sources were used.

Ich versichere, dass ich diese Arbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe.

Heidelberg, 14$^{\text{th}}$ of November 2019 _____

Quirinus Schwarzenböck