Department of Physics and Astronomy
University of Heidelberg

Bachelor Thesis in Physics
submitted by

Moritz Hornung

born in Heidenheim an der Brenz (Germany)

2020

Adapting the Cortical Microcircuit Model for the BrainScaleS-1 Hardware

This Bachelor Thesis has been carried out by Moritz Hornung at the
Kirchhoff Institute for Physics in Heidelberg
under the supervision of
Dr. Johannes Schemmel

Abstract

The neuromorphic hardware BrainScaleS-1 is a wafer-scale analogue system, that is developed at the Kirchhoff Institute for Physics in Heidelberg. Mimicking the brain structure, it is designed to emulate neural networks on an analogue computing substrate. As a means of establishing a benchmark for the system, there currently is an ongoing implementation of the cortical microcircuit model that was proposed by Potjans and Diesmann (2012). A scaled software simulation of this model has already been implemented prior to this work, using the neural network simulator NEST (Gewaltig & Diesmann, 2007). In the course of this thesis, this simulation is adapted to account for the restrictions that are imposed on any model when running on the hardware. This includes a change of the neuronmodel, variation between the individual neuron parameters and moving to network parameters that are realistic with respect to the hardware. A special focus is put on the effects of the synaptic delay, as it has proven to be strongly influential for the firing behaviour of similarly structured networks (Brunel, 2000).

Zusammenfassung

Die Neuromorphe Hardware BrainScaleS-1 ist ein analoges Wafer-Scale System, welches am Kirchhoff Institut für Physik in Heidelberg entwickelt wird. Das System orientiert sich am Aufbau des menschlichen Gehirn und stellt somit eine analoge Platform zur Emulation neuronaler Netzte dar. Gegenwärtig wird das Modell eines ”Cortical Microcircuit” (Potjans & Diesmann, 2012) als mögliche Benchmark für die Hardware untersucht. Dazu wurde mit Hilfe des speziell zur Simulation neuronaler Netzte ausgelegten Programms NEST (Gewaltig & Diesmann, 2007) bereits eine skalierte software Version des ursprünglichen Modells realisiert. Im Verlauf dieser Arbeit wird diese nun schrittweise an die Einschränkungen, welche auf der Hardware gelten, angepasst. Dies beinhaltet unter anderem die Anpassung an ein anderes Neuronmodell, das Hinzufügen von Variaton zwischen den Parametern einzelner Nervenzellen sowie eine Anpassung der für die Simulation wichtigen Parameter an für die Hardware realistische Werte. Dabei wird ein besonderer Fokus auf die Verzögerung des Signals gelegt, da sich diese bereits in anderen Studien als besonders einflussreich für ein ähnliches Netzwerk gezeigt hat. (Brunel, 2000).

Contents

1 Introduction

The human brain can be considered a computing machine with unparalleled efficiency. Due to the limited energy that is available within the human body, it has to be fully functional at roughly 20 watts (Grübel & Baumbach, 2017). Even so, it is able to cope with complex tasks, for example face and pattern recognition, that are considered difficult for modern computers (Chellappa et al., 2010). This extreme adaptation serves as an inspiration for research that is centered around understanding and eventually incorporating the tools that nature has provided us with into modern technologies. As a consequence, modeling and simulating the human brain has become increasingly popular within modern neuroscience (Jordan et al., 2018). However, simulating neural networks on traditional von-Neumann machines becomes very energy consuming, as soon as the scale of the network reaches realistic values with regard to what is seen in nature. Furthermore, it also has the drawback of a huge slow-down when compared to biology (Thakur et al., 2018). This encourages the advancement of alternative systems that circumvent these problems, which is nowadays known as neuromorphic computing.

One approach to this topic is the development of analogue neuromorphic hardware, such as the BrainScaleS-1 (BSS-1) system that has been developed by the Electronic Visions Group at the Kirchhoff Institute for Physics in Heidelberg. This mixed-signal architecture is able to emulate neural networks with a speed-up factor of up to 10000 compared to biological real time, while keeping the energy consumption orders of magnitude below of what is needed for a classical computer to perform the same task (Müller et al., 2020; Schemmel et al., 2010). There currently is an ongoing work of implementing a cortical microcircuit model, proposed by Potjans and Diesmann (2012), on BrainScaleS-1. This neural network, also referred to as cortical column, has recently seen a lot of attention within the field of neuroscience and has established itself as a benchmark (Albada et al., 2018; Cain et al., 2016; Rhodes et al., 2019).

In this thesis, a software simulation of the cortical microcircuit is used to predict the behaviour of the network, when confronted with the restrictions and imperfections of an analogue system, such as BSS-1. To that end, the software simulation is extended to comprise hardware specific effects, including a change of the neuronmodel and variation between the parameters of individual neurons. The simulations are run with the neural network simulator NEST (Gewaltig & Diesmann, 2007), that was also used for the initial publication of Potjans and Diesmann.

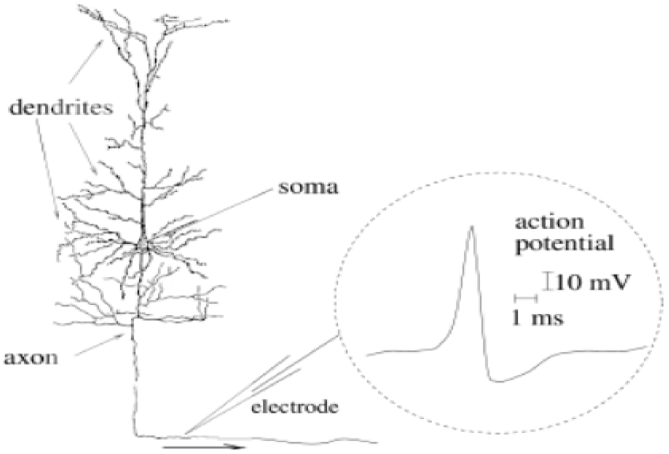


Figure 1: Drawing after Ramón y Cajal. The structure of an individual neuron and the typical shape of an action potential is shown. Figure taken from Gerstner et al. (2014).

2 Theory and Modeling

Neurons are highly specialized cells that act as a foundation for the human brain. As such, they have to acquire information, process it and then forward it to their counterparts. This is made possible by the special structure of these cells, that nature has developed. The concepts that follow are described in depth in the work of Eilers (2019) and Fakler and Eilers (2019), an exemplary look of the neuron structure is shown in Fig. 1.

At the heart of every neuron is its soma (cell body). In addition to providing room for essential components of the cell, such as the cell nucleus or mitochondria, it is also the place where the incoming signals of other cells are integrated. The surrounding membrane gives a clear separation between inside and outside of the cell. This allows for a difference in potential to build up between the inner neuron and the inter cellar medium. Together with an extensive use of ion channels, this enables electricity as a means of data transfer.

The main tool for collecting information from other cells are dendrites. They carry the signal to the cell body, where it is then integrated. Upon reaching a specific threshold voltage, an action potential is generated. The action potential is a fast increase of the membrane voltage that is caused through a temporal change of the membrane permeability towards a specific ion type. It is sent through the axon, the single output of a neuron.

The axon splits up into an axonic tree that is connected to the dendrites of other neurons.

The connection between axon and dendrite is managed through synapses. Upon the arrival of an action potential, neurotransmitters are emitted into the gap between post- and presynaptic neuron, causing ion gates to open and thus transferring the signal chemically.

Understanding the complex interplay that is the result of billions of neurons interacting with each other within the human brain is a huge task and over the years, a lot of work has gone into developing models to help attacking the problem. The models span a wide range of complexity, from a detailed description including several different ion-channels to a simplified integrate-and-fire model. The neuronmodel used throughout this work is that of a leaky integrate-and-fire (LIF) neuron, that will be explained in depth within the next section. A detailed explanation for a variety of different models is found in the work of Gerstner et al. (2014).

2.1 Leaky Integrate-and-Fire Neuron

The leaky integrate-and-fire neuronmodel is a rather simple model, that still succeeds at retaining biological relevance (Petrovici, 2016). In the following, a brief overview of the governing equations and the general functioning of the model will be given.

As the name suggests, the cell membrane is modeled as a leaky integrator. The dynamics of the membrane voltage are hence described as

$$C_{\mathrm{m}}\frac{du}{dt} = g_{\mathrm{l}}(E_{\mathrm{l}} - u) + I^{\mathrm{syn}} + I^{\mathrm{ext}}, \tag{1}$$

where u denotes the membrane potential, C_{m} the membrane capacitance, g_{l} the leak conductance and E_{l} the leak potential. In addition, there is an input current which is divided into a synaptic part I^{syn}, modeling synaptic events, and an arbitrary external part I^{ext}, allowing for additional tuning of the model. Equation 1 is simply the differential equation of a classical RC element from electrodynamics, where we can identify the characteristic leakage ($u \propto -\dot{u}$) and integration terms ($u \propto I$). We can directly see the RC element when taking a look at the analogue circuit of an artificial neuron, shown in Fig. 2.

This simple circuit already does quite well when it comes to modeling the membrane voltage of a neuron with synaptic stimulus, but it does not yet include a spiking mechanism that models the action potential of the cell. Therefore, one has to be added to the model manually. Upon reaching a specific threshold voltage ϑ, the membrane is clamped to a reset potential E_{reset} and held there for the duration of the refractory time τ_{ref}. Only then, it is once again allowed to propagate according to equation 1.

There are two commonly used ways of describing the synaptic input in the LIF model, a current based (CUBA) and a conductance based (COBA) approach. As both of them are of relevance, a short summary will be given in the following paragraphs.

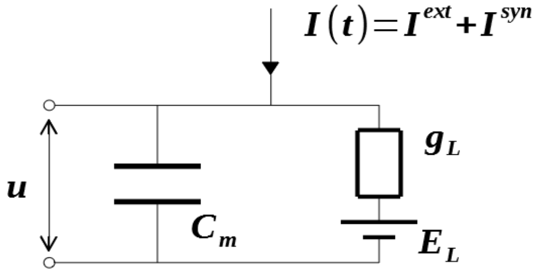


Figure 2: Analogue circuit of a LIF neuron.

In the CUBA case, the main assumption is that most synapses are far away from the soma and thus, what reaches the membrane is effectively a current pulse. The synaptic current is then given through

$$I^{\text{syn}} = \sum_k \sum_s \omega_k \varepsilon(t, t_{s,k}). \tag{2}$$

The first sum is over all of the synapses connecting to the neuron and the second sum over all the spikes with their respective spiketimes $t_{s,k}$. The strenght of a single current pulse is given by the synaptic weights ω_k and the time course by the synaptic kernel $\varepsilon(t, t_{s,k})$. In this model, the synaptic currents of different synapses are independent of each other. The mechanisms of the COBA model stay a bit closer to those of a biological neuron, since the assumption of far away synapses is dropped. A synaptic event is described through a change of the conductance $g_{\text{e/i}}$ towards an excitatory or inhibitory reversal potential $E_{\text{rev,e/i}}$. The corresponding circuit diagram can be seen in Fig. 3. In this case, the current is given by

$$I^{\text{syn}} = g_{\text{e}}(t)(E_{\text{rev,e}} - u) + g_{\text{i}}(t)(E_{\text{rev,i}} - u), \tag{3}$$

where the time development of the conductances is controlled through the synaptic weights and the synaptic kernel:

$$g_{\text{e/i}}(t) = \sum_k \sum_s \omega_k \varepsilon_{\text{e/i}}(t, t_{s,k}). \tag{4}$$

The dynamics of this model are a lot more complex, since the membrane potential appears in the formula for the synaptic current. Because of this, postsynaptic potentials are

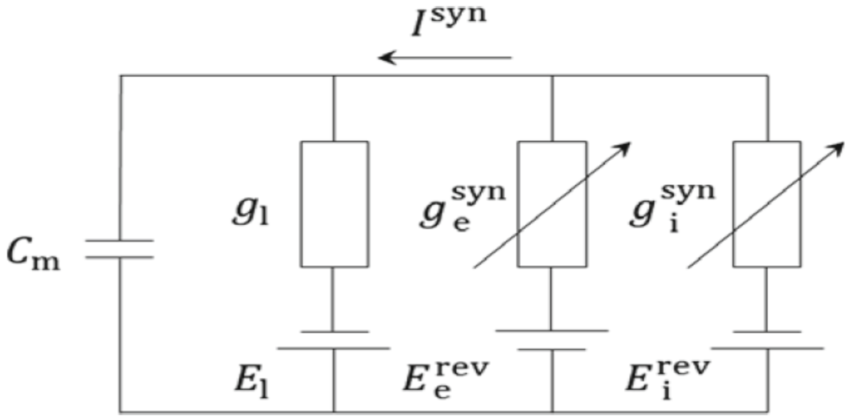

I^{syn}
C_{m}
g_{l}
$g_{\text{e}}^{\text{syn}}$
$g_{\text{i}}^{\text{syn}}$
E_{l}
$E_{\text{e}}^{\text{rev}}$
$E_{\text{i}}^{\text{rev}}$

Figure 3: LIF circuit with COBA synapses. Figure taken from Petrovici (2016).

affected by other synaptic events that arrived shortly before. Another important effect that occurs only in the COBA case is the change of the membrane reaction speed. For the total conductance g of a neuron, the time constant of the capacitor is given as $\tau_{\text{m}} = C_{\text{m}}/g$. Since COBA synapses change the overall conductance of the membrane, this naturally results in a change of the membrane time constant τ_{m}. In the case of a lot of synaptic input, the neuron enters a high conductance state, where the membrane reacts almost instantly to any input.

Finally, the synaptic kernel that is used to describe the time course of the synaptic input in both the CUBA and the COBA model remains to be discussed. A prominent way to describe this is as a difference of exponentials (DoE), with two time constants τ_{rise} and τ_{fall} that correspond to the arrival and removal of neurotransmitters in the synaptic cleft:

$$\varepsilon(t) = \theta(t)\frac{A}{\tau_{\text{fall}} - \tau_{\text{rise}}}(-e^{-\frac{t}{\tau_{\text{rise}}}} + e^{-\frac{t}{\tau_{\text{fall}}}}). \tag{5}$$

Under the assumption that the neurotransmitters arrive very fast ($\tau_{\text{rise}} \to 0$), this model simplifies to an exponential decay:

$$\varepsilon(t) = \theta(t)\frac{A}{\tau_{\text{syn}}}e^{-\frac{t}{\tau_{\text{syn}}}}. \tag{6}$$

As of now, there is no theoretical derivation of this quantity, both of the shown models are purely phenomenological and based on previously measured time constants within a synapse.

2.2 Cortical Microcircuit Model

A local cortical microcircuit, also referred to as cortical column, is a neuron structure that can be found in the early sensoric cortex of mammals. It is thought to be a building block of the brain that is supporting its functionality. Over the years, many different studies concerning this network have been conducted (Beul & Hilgetag, 2015; Cain et al., 2016; Wagatsuma et al., 2011) and thus, it is comparably well known. This thesis builds upon the model that has been established by Potjans and Diesmann (2012). They combined results of anatomical and physiological studies to achieve biological realistic firing rates using the simulation software NEST.

The model, as it was initially proposed, consists of 4 layers that are labeled as L2/3, L4, L5 and L6, with each layer representing a different cell-type of the cortical microcircuit. These then split up into two neuron populations, one inhibitory (i) and one excitatory (e) population. A schematic view of the network is seen in Fig. 4. The main achievement of Potjans and Diesmann was the establishment of a connectivity map connecting the different layers. They extended a single layered balanced random network that was realised with CUBA LIF neurons (Brunel, 2000) and integrated the two main connectivity maps from anatomy and physiology that were obtained in previous studies. A more detailed description of the methods used to derive the connectivity map can be found in the initial

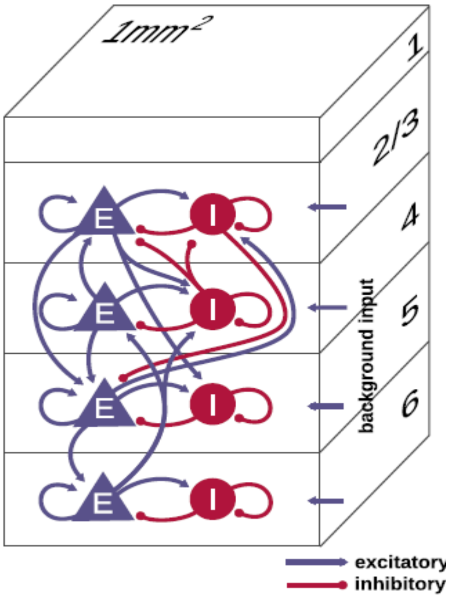
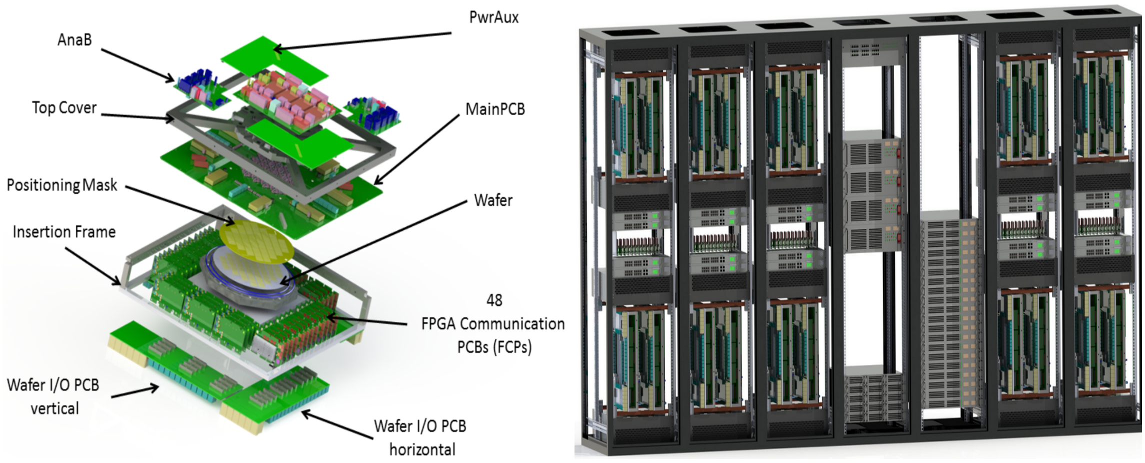


Figure 4: Schematic illustration of the connectivity in a cortical column taken from Albada et al. (2018).



(a) (b)

Figure 5: (a) Overview over a BrainScaleS-1 wafer module. (b) Full system of 20 wafer-modules. Figures taken from the "HBP Neuromorphic Computing Platform Guidebook" (n.d.).

publication Potjans and Diesmann (2012).

Overall, the final model encompasses about 80000 neurons and roughly 3×10^8 synapses, thereby covering $1\mathrm{mm}^2$ of the cerebral cortex. It is a large model and since there are a lot of studies concerning it, it is commonly used as a benchmark (Rhodes et al., 2019).

2.3 The BrainScaleS-1 Wafer-Scale System

Since the goal of this thesis is to end up with a cortical column model that is suitable for the BrainScaleS-1 system, it is essential to understand the mechanisms underlying the hardware and, in particular, to understand the restrictions that it imposes on the model.

BrainScaleS-1 refers to a wafer-scale sized system of neuromorphic hardware that has been developed by the Electronic Vision(s) group at the Kirchhoff Institute for Physics at Heidelberg University. It aims at emulating complex neural systems and has several benefits over its numerical counterparts, mainly when it comes to simulation time and energy consumption (Müller et al., 2020).

A basic building block of the system is the High Input Count Analogue Neural Network (HICANN) chip (Fig. 5 (a)). The HICANN operates based on the exponential integrate and fire (AdEX) model (Schemmel et al., 2010), an extension to the LIF model explained in section 2.1. It is possible to run emulations with the basic LIF model as well, which is going to be the case for the hardware implementation of the column. The chip is using a

mixed signal strategy, where biological neurons are emulated through analogue circuits, but communication between neurons and spike processing is handled digitally. These circuits are built based on the COBA model, forcing a change of the neuronmodel when the cortical column is adapted for the hardware.
Due to the time constants of the circuits being orders of magnitude smaller than those of biological neurons, the emulation runs with a speedup factor of 10^3 up to 10^5 compared to biological real time. A single HICANN already contains 512 neurons with 220 synapses each and is able to emulate small neural networks ("HBP Neuromorphic Computing Platform Guidebook", n.d.).
As a result of the manufacturing process, not all neurons are perfectly identical. This results in an error, that is seen throughout the emulation of any neural network. Since these faults stay the same after the initial manufacturing, this effect is called fixed pattern noise. Even though there is a calibration process that reduces the fixed pattern noise, its existence implies that the neuron parameters on the hardware are not as well defined, as it is the case within a software simulation.
The full BrainScaleS-1 system (Fig. 5 (b)) consists of 20 wafer modules, each one containing 384 HICANNs that can be linked together. In total, this adds up to 196,608 neurons per wafer, with approximately 44 million synapses. The combination of multiple HICANNs allows for the simulation of larger and more complex structures. Even so, the synapse density is not high enough to emulate the full model of section 2.2. This makes the usage of a scaling process that reduces the amount of synapses connected to each neuron in the cortical microcircuit model necessary.

2.4 NEST and PyNN

Moving the network of the cortical column directly to the hardware would prove difficult, as there are quite a few changes to be made to the model. Therefore, a software implementation of the microcircuit is realised (Weidner, 2019), so that the necessary changes can be implemented one after the other. This helps with identifying possible errors in the adjusted column. In fact, since this thesis deals with adapting the model, most of the results that will be described later on were obtained using the software simulation rather than the neuromorphic hardware.
The simulation software used is the neural network simulator NEST (Gewaltig & Diesmann, 2007). This simulator is aimed specifically towards large scale neural networks such as the column and was also used for the original model, making it a reasonable

choice. On top of that, NEST and BrainScaleS-1 support PyNN as API. This guarantees identical neural networks on both platforms and allows for an easier comparison between hardware and software simulation. For running the simulations, NEST version 2.2.2 was used. A more in depth description of the simulator is found within a performance paper by Plesser et al. (2007).

PyNN (Davison et al., 2009) is a popular Python application that simplifies the usage of different neural network simulation tools. It provides a common interface so that a single network code can be run on different simulators or even on neuromorphic hardware.

2.5 Network Behaviour

The classification of the firing behaviour that is used throughout this thesis is motivated through the work of Brunel (2000). However, the formulas used to obtain quantitative values are taken from Potjans and Diesmann (2012).

The first quantity, describing the irregularity in the spiking behaviour of a single neuron, is the coefficient of variation (CV) of its interspike interval (ISI). For any given neuron, the ISI is defined as the time between two outgoing spikes. In general, this is not a fixed value but a distribution P_{ISI}. The CV is then given as the standard deviation normalized to its mean value:

$$CV(P_{\text{ISI}}) = \frac{\sigma(P_{\text{ISI}})}{\mu(P_{\text{ISI}})}. \tag{7}$$

This measure is easily expanded to describe the irregularity of a neuron population, simply by averaging over the CVs of all the individual neurons. A perfect Poisson process has a CV value of one, whereas for regularly firing neurons, the value converges to zero. Throughout the following sections, irregularity always refers to the averaged CV of the ISI.

The second quantity, used to describe the global behaviour of neurons within a population, is the synchrony. The accounted time interval is split into bins and the number of neurons that spiked per bin is counted. The synchrony is then given as the variance of the resulting spike count histogram divided by its mean. This is not an exact quantity as it depends on a the binwidth of the spike count histogramm, but the order of magnitude gives a general idea of the neuron behaviour. Large synchrony values indicate that the neurons fire at the same time, smaller values indicate that there is almost no corellation between the spiketimes of the neurons.

With the help of these quantities, the firing behaviour categorizes a network into four different regimes, that were originally declared by Brunel (2000):

- A synchronous regular (SR) state is characterized through large synchrony and small irregularity values. In this state, the neurons within a population are divided into a few clusters, with the neurons that belong to the same cluster fring at the same time. These clusters then behave as oscillators with a regular time interval between the spikes.

- In the asynchronous regular (AR) case, synchrony and irregularity of the network are small. Individual neurons fire at a constant rate but they do so independently from other neurons in the population.

- The synchronous irregular (SR) state exhibits the same oscillatory global behaviour that was observed for the SR state, however with varying time spans in between the spiking. Synchrony and irregularity are large.

- Finally, in the asynchronous irregular (AI) state, individual neurons show strongly irregular firing with no obvious correlation to other neurons. This is represented through a small synchrony value with high irregularity. This state was observed for the original model in the publication of Potjans and Diesmann (2012).

3 Experiments and Results

Adapting the cortical microcircuit model to the BrainScaleS-1 hardware is a large project and outside of this thesis, within the work of Weidner (2019) and Schwarzenböck (2019), a lot of effort has already been put into it. Of the problems that were touched on, when introducing the hardware in section 2.3, the transition towards COBA neurons and the effects of parameter variation between individual neurons will be discussed in depth throughout this work. A reimplementation of the column, using the PyNN interpreter, as well as the scaling problem were already taken care of and built upon when investigating further througout this thesis. However, since the scaling process is not trivial, a short summary of the problem and its solution follow.

Because of the limited number of synapses that are available on the BSS-1 wafer-scale system, the model needs to be scaled down to about ten percent of its initial size. Unfortunately, even at ten percent, keeping the ratio of synapses per neuron the same as in the original column is not feasible. The amount of synapses needed for this would exceed what is possible on the hardware. To bypass this problem, the indegree (the amount of synapses connected to a neuron) is decreased, while simultaneously increasing the weights of the remaining synapses. This is done to keep the same synaptic input to the neuron with less synapses connecting into it. As a downside, the input variance of the network grows, since a single neuron is much more influential. To balance some of this effect, the external poisson input used in the model of Potjans and Diesmann (2012) is substituted with a DC input. The scaling process was motivated through the work of Albada et al. (2015), where the impact it has on mean firing rates and correlation within a neural network is described in detail.

The downscaling leads to a slight deviation in the mean per population firing rates when compared to the full column. To keep this effect as small as possible, the two multiplicative scaling parameters ”exc” and ”inh” were added. They scale the excitatory and inhibitory weight of all the populations and provide an additional tuning point for the model. Doing so, they allow to optimize for a minimal difference between the firing rates of the full model compared to those of the scaled column. The results of this optimization are shown in Fig. 6, where the mean firing rates of the self implemented model are plotted together with the values that Potjans and Diesmann (2012) found in their study.

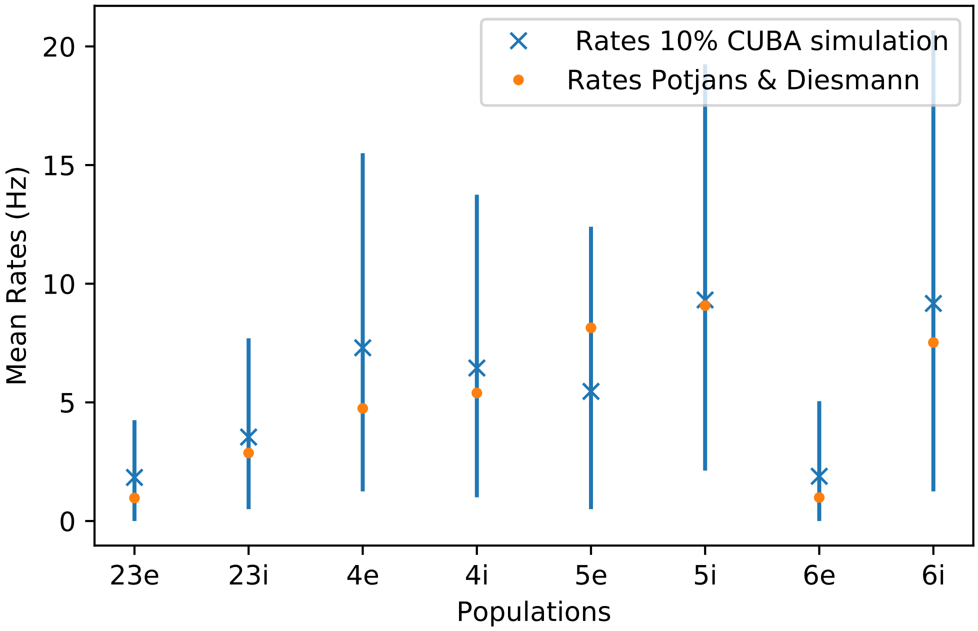


Figure 6: Mean per population firing rates of the downscaled 10% CUBA model that served as a starting point. The additional scaling parameters are set to exc 0.5 and inh 1.5 and the simulation was run up to 5000 ms with a time step of 0.1 ms. The first second of the simulation was excluded, allowing the network to stabilize. The error bars give an insight over the spread that the firing rate has between the different neurons within a population. This is expected and not to be confused with the uncertainty of the mean value that can be found for different seeds of the network.

The neuron parameters that are used for the simulation are taken over from the initial model (Potjans & Diesmann, 2012) and are summarized in Table 1. Unless specified otherwise, all the results presented from here on were obtained using a time step of 0.1 ms and simulating up to 5000 ms. Spikes within the first second are excluded from the results, as the network needs time to stabilize.

Table 1: Summary of the parameters used for the CUBA simulation.

Name	Value	Description
C_m	0.25 nF	Membrane capacitance
E_rest	-65 mV	Leakage potential
E_reset	-65 mV	Reset potential
ϑ	-50 mV	Firing threshold
τ_m	10 ms	Membrane time constant
τ_ref	2 ms	Refractory period
$\tau_\mathrm{syn,e/i}$	0.5 ms	Synaptic time constants
D_e	1.5 ms [1]	Delay for excitatory connections
D_i	0.8 ms [1]	Delay for inhibitory connections
ω	87.8 pA [2]	Excitatory synaptic weights
g	-4	Relative inhibitory weight

[1] : Values differ for every connection and are drawn from a normal distribution around the given value with a standard deviation of 50%

[2] : Distributed in the same way as the delays with a standard deviation of 10%. The weights are influenced by the scaling parameters exc and inh and the connection from populations 4e to 23e is multiplied with an additional factor of 2

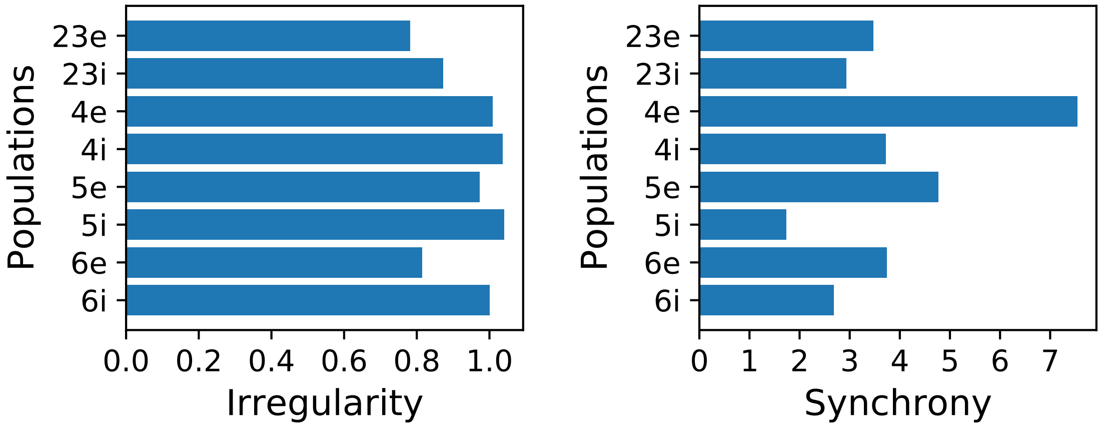
For further discussion later on in the thesis, synchrony and irregularity of this scaled version of the CUBA column are depicted in Fig. 7. With irregularities slightly below one and the synchrony being of the order of magnitude 1, the asynchronous irregular firing behaviour of the initial column is not lost during the scaling process, despite an expected increase in the correlation within the network (Albada et al., 2015).

3.1 Transition to COBA Neuronmodel

Seeing that the scaling problem has already been addressed, the next thing to investigate is whether the cortical column withstands a transition to the COBA neuronmodel used on the hardware. Even though large parts of the model are the same, the dynamics of the CUBA LIF model are not equivalent to those of the COBA LIF model and there is no such thing as a map that can be applied every time to switch between the models.

For a first attempt, we try to keep the synaptic current constant when changing the synapse model. Comparing the corresponding equations from section 2.1 yields the following connection between the weights in the CUBA case and those in the COBA case:

$$g_\mathrm{e/i} = \frac{\omega_\mathrm{e/i}}{E_\mathrm{rev,e/i} - u}. \tag{8}$$



(a)

(b)

Figure 7: (a) Irregularities of the populations in a 10% CUBA column. Values close to 1 indicate a wide distribution of the interspike interval. (b) Synchrony for the same network. The binwidth of the spike count histogram is 3 ms.

To distinguish between the synaptic weights of both cases, the COBA weights have been relabeled to $g_{\mathrm{e/i}}$. On first glance, the upper equation seems quite simple. However, there is an explicit occurrence of the membrane potential that needs further treatment.

For COBA LIF neurons in the high conductance state, it can be shown that replacing the potential with its mean value is a reasonable approximation. A mathematical sound proof of this can be found in the thesis of Petrovici (2016). Because the connection density of the model is high, this can also be applied to our case, which simplifies the problem considerably. The task at hand is now to find an estimate for the mean membrane potential in the CUBA case.

To begin with, the known firing rates r_{mean} together with the average amount of synapses connected to a neuron K are used to estimate the rate at which synaptic events happen:

$$r_{\mathrm{syn}} = K * r_{\mathrm{mean}}. \tag{9}$$

Furthermore, the charge transfer per synaptic event is given through integration of the synaptic current as

$$Q_{\mathrm{e/i}} = \tau_{\mathrm{syn}} * \omega_{\mathrm{e/i}}. \tag{10}$$

Combining both of these quantities yields the mean synaptic current that goes into a neuron:

$$I_{\mathrm{mean,e/i}} = \tau_{\mathrm{syn}} * \omega_{\mathrm{e/i}} * K * r_{\mathrm{mean}}. \tag{11}$$

As an estimate for the mean membrane potential without any synaptic input, we can simply use the leakage potential E_l. Adding the mean inhibitory and excitatory currents on top of it will thus give an estimate for the mean membrane potential with synaptic input:

$$E_{\mathrm{mean}} = E_{\mathrm{l}} + \frac{\tau_{\mathrm{m}}}{C_{\mathrm{m}}}(I_{\mathrm{mean,e}} + I_{\mathrm{mean,i}}). \tag{12}$$

The above equations provide a way to transition the synaptic weights. However, to complete the transition of the neuronmodel, we still need to choose fitting reversal potentials for the conductance based synapses. As the absolute value of these does not directly affect the simulation and only the difference to the membrane potential is important, they were set symmetrical around the threshold potential to -100 mV for the inhibitory and 0 mV for the excitatory reversal potential. With a range of 100 mV, it is also safe to be supported by the hardware, as for example in the “HBP Neuromorphic Computing Platform Guidebook” (n.d.), a range of 160 mV is used.

With this, all the necessary parameters for the COBA model are set. The ensuing firing rates, already combining the scaling process and the transition to the COBA neuronmodel, are shown in Fig. 8. Comparing with the rates of the scaled CUBA simulation shows only small differences, thereby justifying some of the approximations used to derive the weights in hindsight. The main deviation from the target rates manifests within populations 4e and 5e, the former firing at a increased rate and the latter at a decreased rate. The difference for both of them is roughly 2.5 Hz. To further improve the results, the thresholds of the different populations could be adjusted manually, thereby fine tuning the rates of single populations. However, the goal for the software simulation is only to produce a model that shows rates within the correct order of magnitude, as many things will have to be changed again when actually moving to the hardware emulation. Therefore, no adjustments were undertaken as of yet.

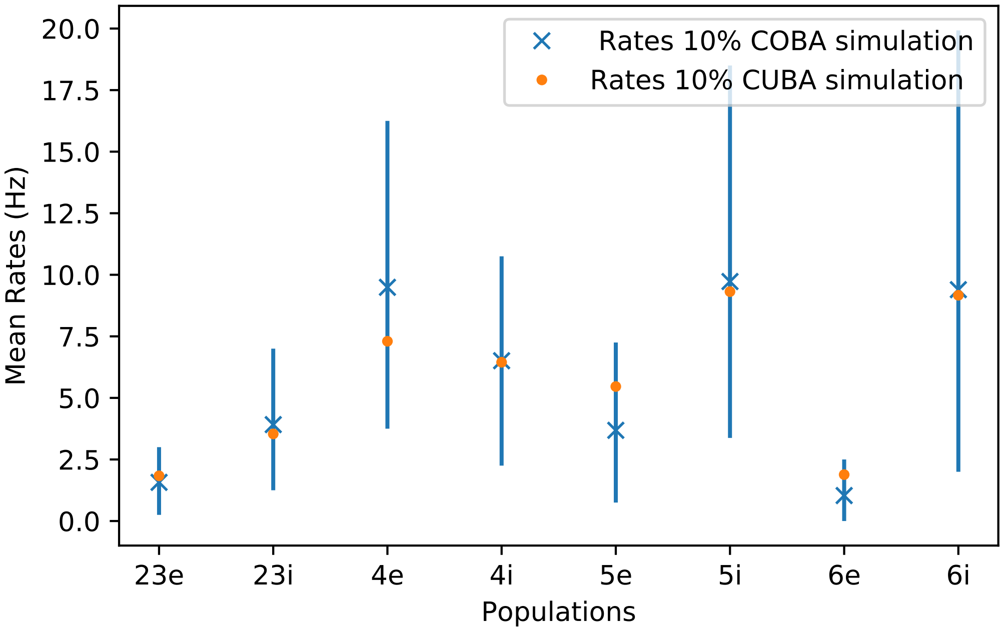
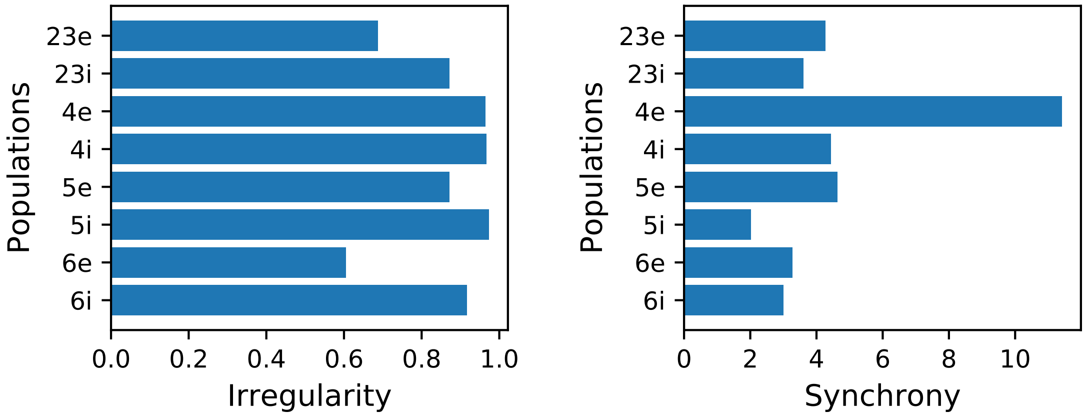


Figure 8: Mean per population firing rates after transitioning to the COBA neuronmodel. The scaling parameters are set to exc=0.7 and inh=1.4. Compared to the rates of the 10% CUBA column, we find an increased firing rate in the 4e population and a decrease for the 5e population.

Finally, the impact that the transition has on irregularity and synchrony of the model is depicted in Fig. 9. The irregularity shows a strong influence for population 6e, where the CV of the interspike interval drops to 0.6. Within the other populations, the change is a lot less pronounced, but the trend is always towards smaller values than seen for the 10% CUBA simulation. Some of these changes might be within the range of statistical variation, but it has already been shown that the mean rates in the CUBA case are very stable (Weidner, 2019) and further testing with differently seeded random number generators also shows the stability of mean rates, irregularity and synchrony for the COBA model. Unfortunately, there was not enough time to quantify this process. Together with the fact that all of the deviations are towards smaller values, it is safe to assume that they are caused by the change of the neuronmodel and do not just fall under statistical variation due to different weights being drawn. The same argumentation also holds for the synchrony, where an increase is observed for the COBA model.



(a)

(b)

Figure 9: (a) Irregularity for the 10% COBA model. The change of the neuronmodel causes the irregularity values to decrease. This effect is strongest for the 6e population. (b) Synchrony for the 10% COBA model. Synchrony values go up for COBA synapses.

3.2 Simulation with Distributed Parameters

The next change applied to the model is to account for the fixed pattern noise that is unavoidable on any analogue system. Despite a calibration, that reduces the strength of this effect, there will always be some variation of the neuron parameters as a result of the manufacturing process. On top of that, there is a statistical component to this variation due to thermal noise as well. To investigate the behaviour of the model under these circumstances, we want to include parameter disparity between neurons within the NEST simulation.

We assume that the neuron parameters are normal distributed around the initial values, with a relative standard deviation of 10% for any voltage and 20% for the time constants. These values were taken from Schwarzenböck (2019), as a first estimate of what is expected on the hardware.

In case of the reversal potentials, applying the relative standard deviation directly would be incorrect, since the absolute value of the excitatory potential is close to 0 V. In the end, only the difference between reversal and membrane potential is relevant for the system, so this difference should be affected with the uncertainty. Because the mean membrane potential is assumed to be close to the leakage potential, we can apply the relative deviation to the difference between reversal and leakage potential to obtain a value for the width of the gaussian. This value is then taken as variation to the reversal potential, which

Table 2: Parameters of the normal distributions used to account for the fixed pattern noise.

parameter	μ	σ	boundaries
C_m	0.25 nF	5%	$(0, \infty)$
E_l	-65 mV	10%	$(-\infty, \infty)$
ϑ	-50 mV	10%	(-65, ∞)
E_reset	-65 mV	10%	$(-\infty, \vartheta^1)$
$E_\mathrm{rev,e}$	0 mV	10% [2]	$(-\infty, \infty)$
$E_\mathrm{rev,i}$	-100 mV	10%[2]	$(-\infty, \infty)$
τ_m	10 ms	20%	$(0, \infty)$
τ_ref	2 ms	20%	$(0, \infty)$
$\tau_\mathrm{syn,e/i}$	0.5 ms	20%	$(0, \infty)$

[1] : Boundaries for the reset potential are calculated after all the threshold values have been drawn to avoid $E_{reset} > \vartheta$.

[2] : Applied to the difference between reversal and leakage potential.

translates directly to a variation of the difference. A summary of the distributions used for the parameter variation is given in Table 2.

Additionally, we have to prevent unphysical values from being drawn out of the random distributions. By setting according boundaries and redrawing the values that are on the outside, this is easily taken care of for most of the parameters. However, threshold and reset potential need a special treatment, as we also have to avoid a reset potential larger than the threshold. This constellation would lead to neurons that fire at intervals of their refractory period, which is not desired. It is circumvented by adapting the boundaries for the reset potential after the threshold values have been set. This treatment shifts the mean value of the reset potential by about 5V, when variation is applied to both reset and threshold at the same time, but this does not largely affect the outputs of the simulation.

The results for individually affected parameters are shown in Fig. 10, where the difference between the firing rate with and without variation is plotted. The final data point shows the influence of the variation, when applied to all parameters at the same time. The measurement is averaged over ten different seeds for the random number generator that generates the respective distributions.

For most of the parameters that have been tested, the effects are negligible. Especially the influence of variation on the time constants and reversal potentials seems to be almost vanishing. On the other hand, a strong influence is observed for the threshold and leakage potential, which is reasonable, considering that they are the main tuning points used when adjusting the firing behaviour of LIF neurons.

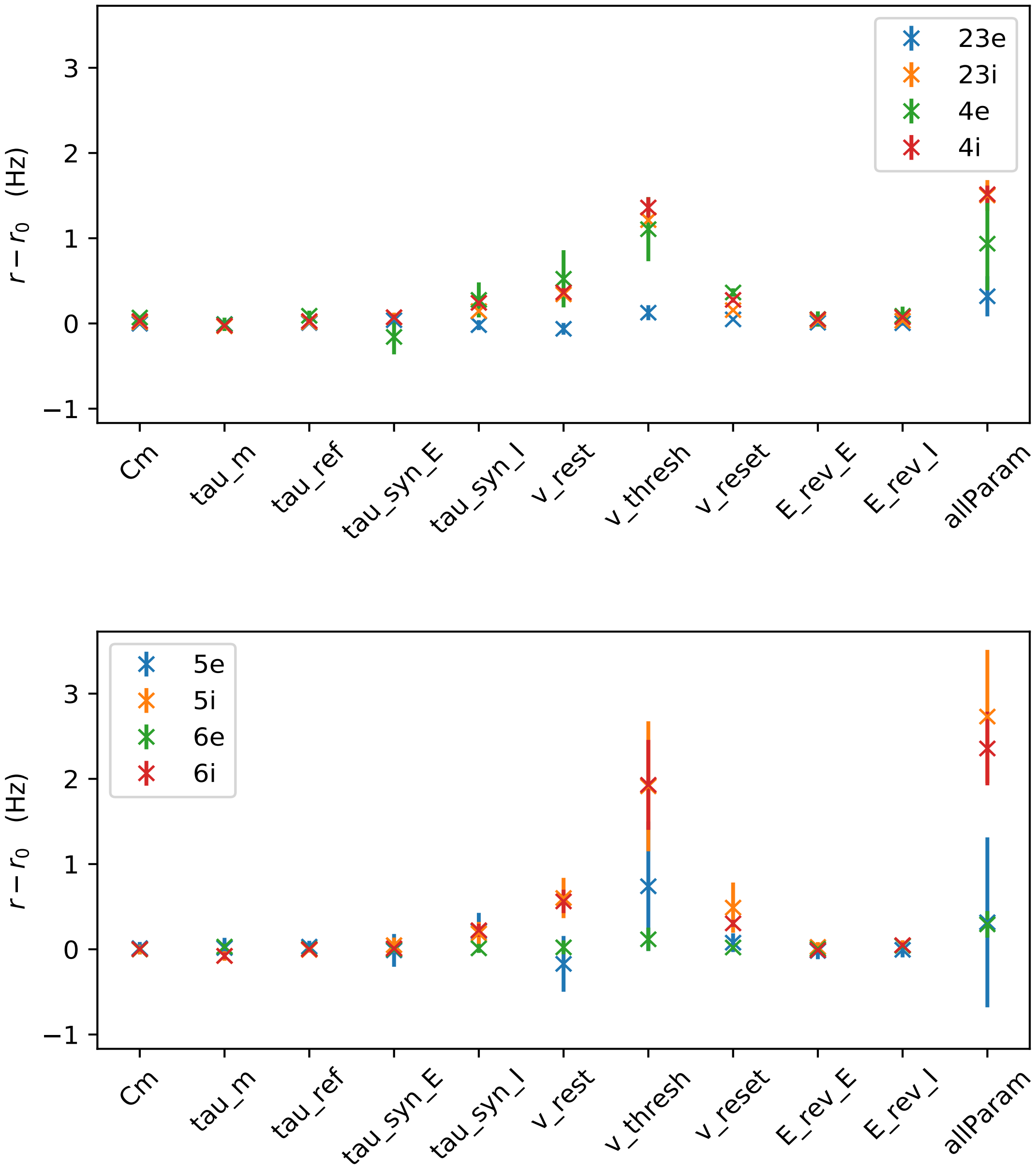
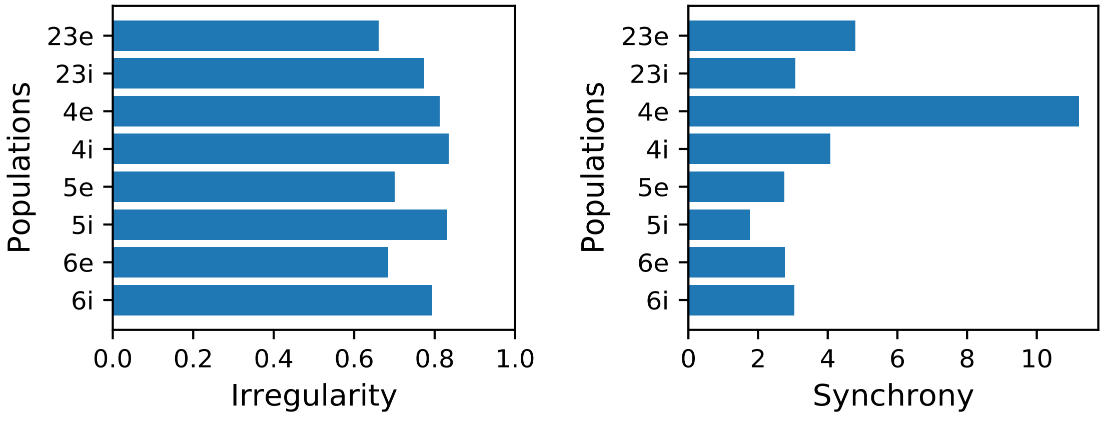


Figure 10: Parameter variation between individual neurons. The difference between the mean firing rates of the 10% COBA column with and without noise on the neuron parameters is plotted. The variation is in turn applied to the different parameters. The 'allParam' datapoint stands for a measurement where all the parameters are affected at once. To account for statistical effects, the measurements are repeated for ten different seeds of the RNG that generates the variation and the averaged deviation is plotted.



(a) (b)

Figure 11: Exemplary plot for a 10% COBA simulation with variation on all parameters according to Table 2. (a) Irregularity values. A trend towards smaller values is observed compared to the simulation without variation. Overall, it is more balanced with the irregularity of the 6e population being slightly larger and a lot closer to the others. (b) Synchrony values. The difference to the case without variation is negligible.

In general, the effect causes an increase of the firing rates for almost all of the parameters. A possible explanation for this trend might be the strong correlation between the neurons, that is established within the scaling process. Within the downscaled model, individual neurons have more influence, to a point where a single neuron that spikes at a high frequency might be reflected in the total behaviour of the network. This assertion could be validated through the simulation of a full scale COBA model with parameter variation, that so far has not been realised.

Another interesting observation is that for some populations, for example 5e, the effects of the variation of only the threshold is stronger than varying all of the parameters. This implies that additional uncertainty on some parameters stabilizes some properties of the model rather than destabilizing it. This seems reasonable when remembering the biological background, where no neuron is exactly the same and yet we have a stable firing pattern.

The impact that the change has on synchrony and irregularity is plotted exemplary in Fig. 11, with all parameters affected by variation, as this is the expected state on the hardware. Overall, we find a small decrease in the irregularity with most of the populations showing a CV around 0.8. However, the behaviour is now more consistent throughout the different populations, as the value of population 6e increased slightly and is therefore not so far from the others. The synchrony does not change notably, when compared to the simulation outputs without noise.

3.3 Synaptic Delay Investigation

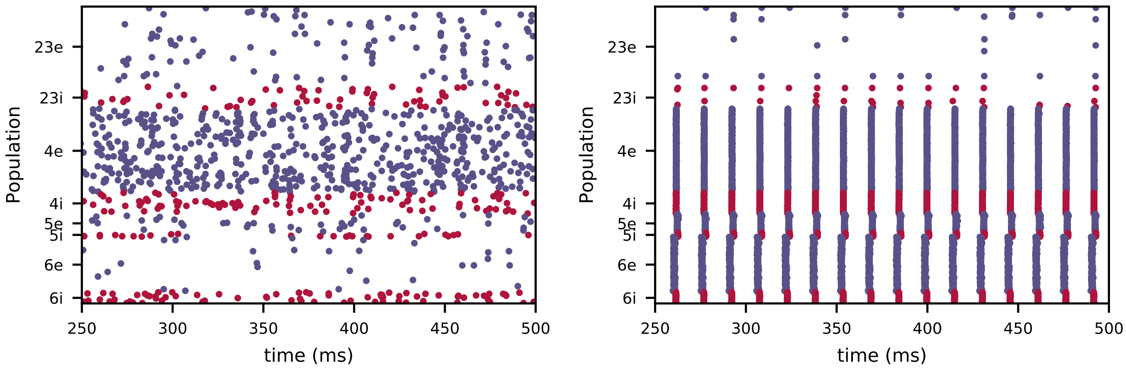
So far, mean firing rates as well as the second order statistics of the model, synchrony and irregularity, are largely preserved throughout the different changes. However, up to this point, the synaptic delay has not been considered. The delay describes the time that passes between the emission of a spike and the arrival of the signal at the receiving neuron. In principle, it could have been part of the section regarding the parameter variation. However, since it is expected to have a strong influence on the firing pattern of the network (Brunel, 2000), it was separated from the other changes for a more thorough investigation. The previously implemented parameter variation is turned of for the investigations in this section.

3.3.1 Adapting the Model

As a first step, the delay specific parameters used by Potjans and Diesmann (2012) were adjusted to be more realistic with regard to the hardware system. For now, this means that the mean delay for inhibitory connections was increased to match excitatory connections at 1.5 ms. The underlying distribution was changed to being uniform, with a relative broadness of 50% in both directions. This is a first guess of what we expect for the hardware. Estimating realistic delay values proves to be a difficult task that will be addressed later on, in section 3.3.3.

A raster plot of the spiketimes for the adjusted network, together with a reference plot of the behaviour previous to the change, is seen in Fig. 12. For every spike that a neuron emits, a dot is created at the respective time. To keep the amount of data manageable, only ten percent of the neurons within every population are depicted. The reference plot (Fig. 12 (a)) shows the unperturbed asynchronous irregular state that the cortical column exhibits in section 3.1. For the second plot, the delay values are distributed between 0.75 ms and 2.25 ms for both connection types. With these settings, the system fully synchronizes with almost all of the neurons firing at the same time. It can be seen that these excitations are periodic, displaying a fixed time interval between two neighbouring events.

The plots emphasize the transition of the firing pattern that the model has undergone. The



(a)

(b)

Figure 12: Raster plot of the spiketimes for a time window of 250 ms. The simulation is started at 0 ms. (a) Plot of the asynchronous irregular behaviour before the delays are adjusted. (b) Plot after the adjustment. The underlying delay distribution is now uniform between 0.75 ms and 2.25 ms. The system is fully within a synchronous regular state

asynchronous irregular firing that is usually expected within a cortical column is lost. This is also reflected in the quantities used to describe the firing patter, where a strong increase in the firing rates and synchrony is found. At the same time, the irregularity values drop to almost zero. The system stabilizes within a synchronous regular state, where almost all of the neurons fire at the same time. Testing for other parameters of the distribution yields similar results for a wider spread of the delays, as well as for a larger mean value. To determine where the problem is located, the simulation is run with the same parameters for the CUBA neuronmodel. This shows a cortical column in the expected asynchronous irregular state, indicating that the problem lies within the change of the neuronmodel. Taking a look at the transition described in 3.1, the main adjustments were made to the synaptic weights and the reversal potentials. Further measurements also show that, with values close to -85 mV for multiple populations, the mean membrane potential of the CUBA case is very close to the inhibitory reversal potential of -100 mV, that was chosen in section 3.1. This effectively means a weakening of the inhibitory spikes, since they are, in a way, weighted by the difference between membrane and reversal potential. To solve this problem, the reversal potentials were set from -100 mV to -150 mV for the inhibitory and from 0 mV to 50 mV for the excitatory potential, thereby reducing the effect that was described above. With these new reversal potentials, the column is able to stabilize in the asynchronous irregular regime. However, for different parameters of the delay distribution and differently seeded random number generators, there are still combinations that lead to the synchronous firing behaviour that was observed earlier. A more detailed investigation of this development follows in the next section.

3.3.2 Influence on Network Behaviour

The effects of delay variation on a balanced random network based on CUBA neurons have already been documented in the work of Brunel (2000). Even though the layers of the column are not identical to the network that was investigated in this publication, they closely resemble it. Keeping this in mind, the findings of Brunel should give a good indication of what to expect for the cortical microcircuit. In the paper, both the mean value and the spread of the distribution are shown to influence the firing pattern. To see how these effects carry over to the cortical column model, the behaviour it demonstrates for different combinations of mean value and spread of the delay distribution is examined. For this measurement, the mean delay values are varied between 1.5 ms and 3.5 ms in steps of 0.25 ms and the relative width of the distribution from 20% to 80% in steps of 10%. To account for the statistical component of the delay values, the measurements were repeated for 10 different seeds of the random number generator used for the distribution. In the area where the transition between the firing patterns is observed, from 2.0 ms to 2.75 ms, the amount of seeds was increased to 40, to obtain a better statistic. For a given seed, the network with delay adjustments is compared to the state of the network at the end of section 3.1, with no other parameter variation applied. It is considered to be in a synchronous state, when the sum over all populations of the difference between the mean firing rates is larger than 20 Hz. Fig. 13 shows the resulting probability of finding the network in a synchronized state, depending on the mean delay value and the broadness of the distribution. There is a sharp border between delay values that show a synchronization and values that do not, however, the exact point at which the firing pattern transitions also depends on the width of the distribution. Together with the dependence on the reversal potential found in the previous section, the results can be summarized to three main observations:

- There is a critical delay, after which the network will always show synchronous firing behaviour.
- This critical point also depends on the distribution width, for a larger distribution it gets shifted to larger delay values.

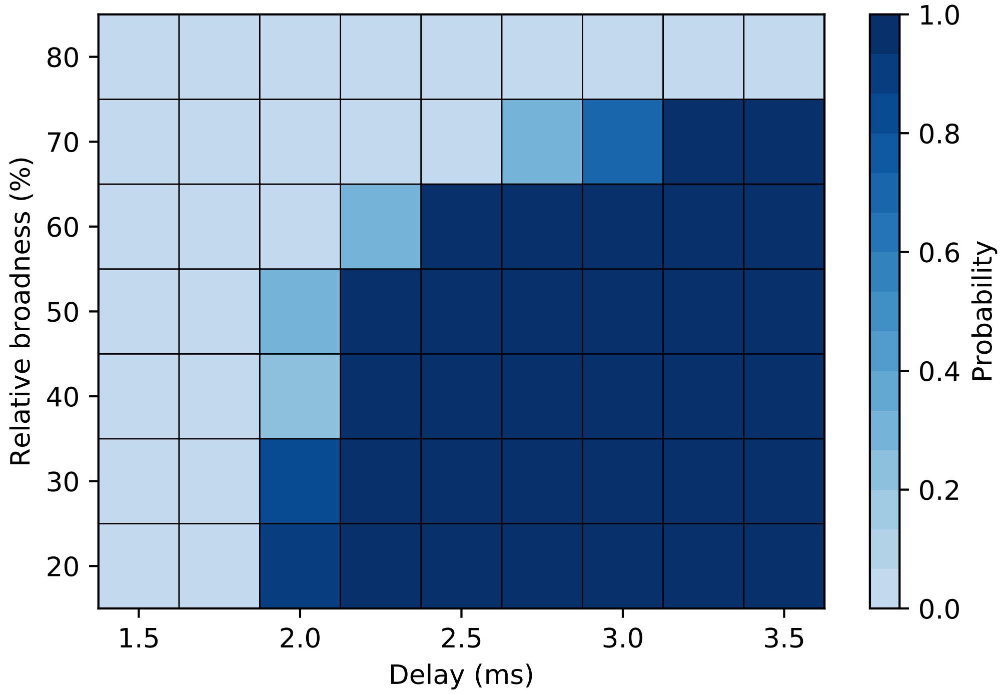
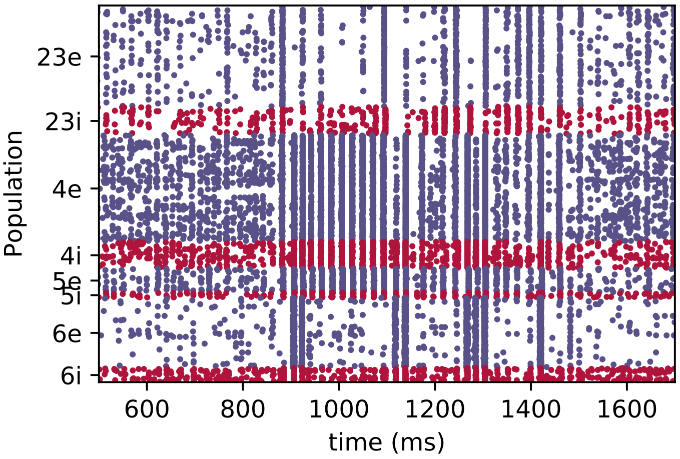


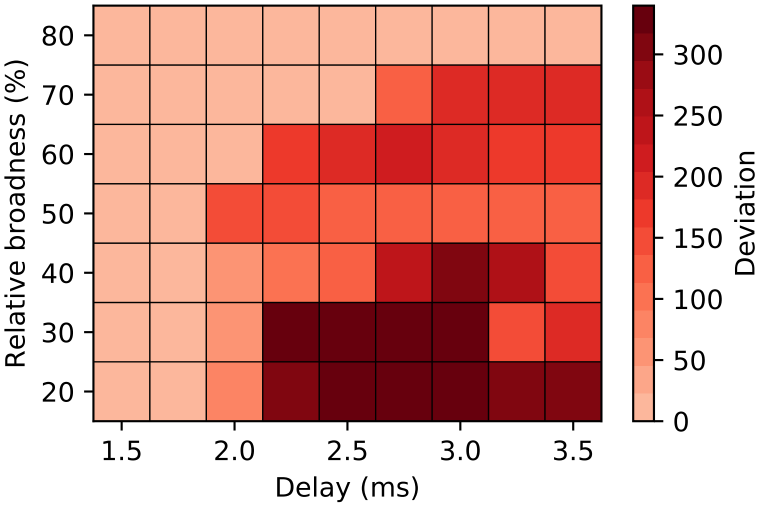
Figure 13: Probability distribution for the occurrence of a synchronization within the firing behaviour of the 10% column with adjusted delay values. The likelihood of a synchronized state increases with higher mean values but decreases with the distribution width. The data was obtained through the use of different seeds for the RNG that generates the delay distributions.

- A larger dynamic range of the membrane realised through a larger distance between the reversal potentials also decreases the probability of synchronized firing.

The influence of the distribution width was already observed for the CUBA model (Brunel, 2000), where a distributed delay prevented any synchronous activity of the balanced random network. This also seems to be applicable in the COBA case, since an increase in the distribution width caused the network to stay in the asynchronous regime.
The effects of the reversal potential can also be explained through the findings of Brunel. With the membrane potential being close to the inhibitory reversal potential, inhibitory spikes have less impact. This can be seen as a reduction of the effective inhibitory weights, thereby shifting the ratio between inhibitory and excitatory weights. In the research of Brunel, a smaller ratio was observed to produce a firing pattern in the synchronous regular state.



(a)



(b)

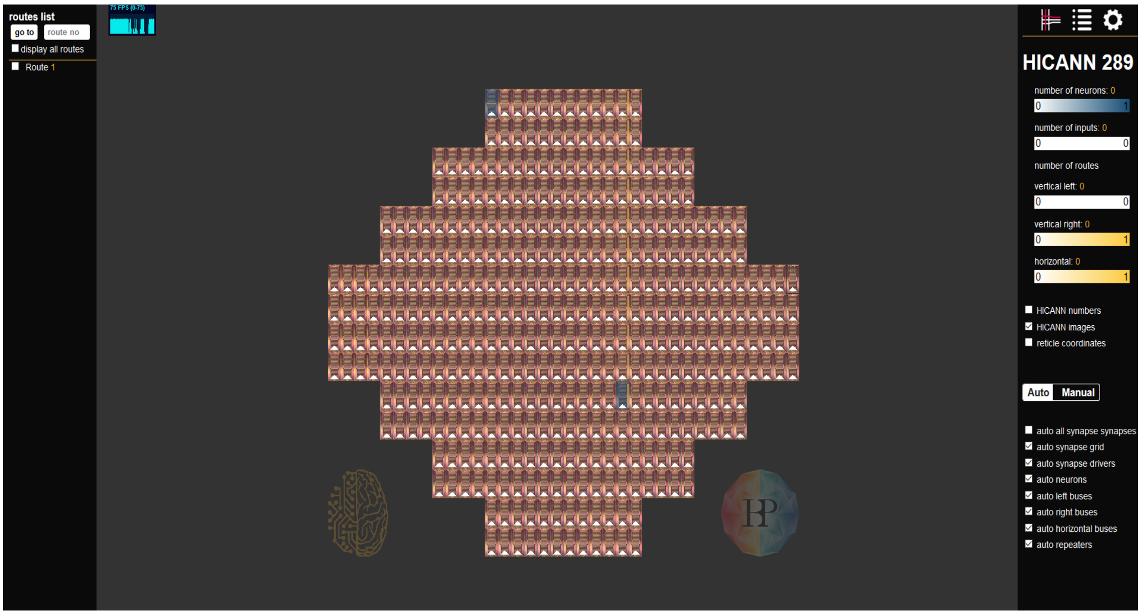
Figure 14: (a) Exemplary raster plot for an oscillating state. The network fully synchronizes around 900 ms and then looses some of the synchronization at roughly 1400 ms. (b) Illustration of the strength of the synchronization process. The mean deviation to the firing rates of the undisturbed model is plotted depending on the distribution width and mean value. It is averaged over all seeds that show a synchronization.

There is one more interesting observation that does not become apparent from Fig. 13. For many combinations of the parameters for the delay distribution, the network is able to oscillate between the synchronous and the asynchronous state. This also causes an increase of the mean rates, albeit only a small one when compared to the rates of a fully synchronized network. Such a transition is seen in Fig. 14 (a), where the system synchronizes roughly after 0.9 ms and then loses the synchronization, going back over to an asynchronous state at 1.5 s. This results in a variety of different values for the mean rates of the system, as the effect can happen with different intensity. To give a better overview of the strength of the synchronization within the initial plot of the probabilities (Fig. 13), a second plot is created, where the total deviation to the mean rates is shown instead of the probability. Fig. 14 (b) emphasizes the strength of the synchronization process. It shows that an increased distribution width does continually reduce the strength of the synchronization, that is maximal for a relative broadness of 20%. It is also interesting to see that the deviation in the mean firing rates is strongest for delay values between 2 ms and 3 ms, delay values that are even larger appear to have a weaker influence on the network. However, this decrease is probably a direct consequence of the increased synaptic delay, since it simply takes longer for the signal to propagate to the next neuron, resulting in lower firing rates. It does not mean a lessening of the synchronization within the network.

3.3.3 Estimated Hardware Delays

Within the previous section, a general study of the impact that different delays have on the model was conducted, disregarding the actual values that are given on the hardware. It is difficult to theoretically predict realistic values for the analogue system, since a lot of different factors play into the signal propagation and have to be considered. Nonetheless, a good estimate might be useful to be able to evaluate the possibility of a hardware implementation.

Within this section, an empirical approach is taken with the goal of obtaining an estimate through the emulation of a very basic network and the measurement of the reaction time. For this purpose, a network containing only two neurons is placed on different positions of the wafer. The mapping process that handles the placement is automated, and a web visualization displays the outcome. Fig. 15 shows an example of such a mapping, with the two neurons being placed on different HICANNs of the wafer that are highlighted blue. The route along which the signal is propagating is also visualized, but unfortunately only barely visible on the plot, as it is a light yellow. It goes from the top left in a line



□ display all routes

□ Route 1

HICANN 289

number of neurons: 0

0 1

number of inputs: 0

0 0

number of routes

vertical left: 0

0 0

vertical right: 0

0 1

horizontal: 0

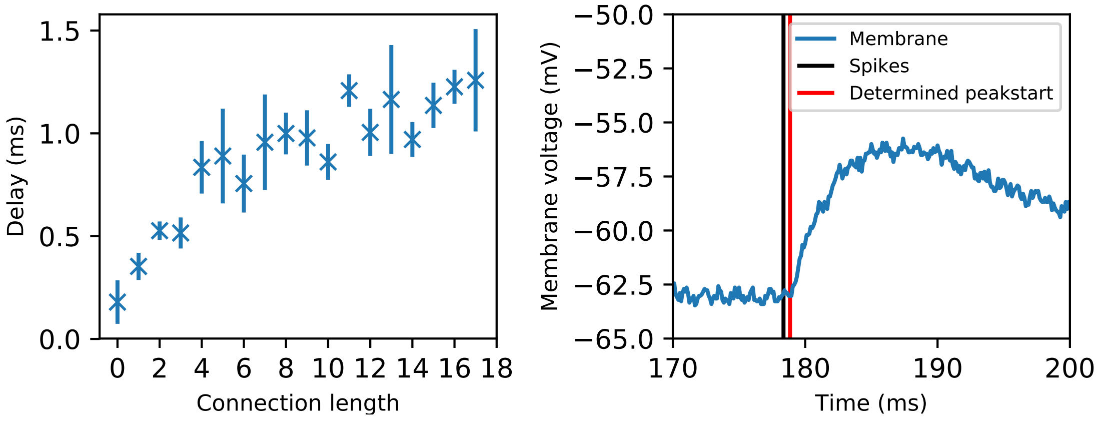
0 1

□ HICANN numbers
☑ HICANN images
□ reticle coordinates

□ auto all synapse synapses
☑ auto synapse grid
☑ auto synapse drivers
☑ auto neurons
☑ auto left buses
☑ auto right buses
☑ auto horizontal buses
☑ auto repeaters

Figure 15: Web visualization of the mapping for the two neuron network that is used to estimate the delays. The rectangles present the available HICANNs on the wafer. The two HICANNs in use are highlighted in blue.

to the right and when it reaches the same position on the x-axis as the second HICANN, it goes straight down. The first neuron is setup with a leak over threshold potential, causing it to fire periodically without the need of additional external stimulaton. The second neuron is set up with typical parameters that were also used for the neurons within the cortical column to create a scenario that is as close to the final network as possible. This should however not affect the delays. The time between two spikes can easily be adjusted through the refractory period of the presynaptic neuron, guaranteeing that the postsynaptic neuron has returned to the leakage potential before a second spike arrives. A projection between the neurons is created, using a PyNN AllToAllConnector. This connector creates a connection between all neurons of the first population to all neurons of the second population. In our case, this is exactly one connection. To make the signal as strong as possible, the weights are set to the maximum value that is possible on the BSS-1 hardware, setting the hardware specific parameters to a digital weight of 15, a gmax factor of 1023 and a gmax_div factor of 2. The spiketimes of the presynaptic and a membrane trace of the postsynaptic neuron are recorded. The delays are then determined as the difference between the spiketime of the first neuron and the peakstart within the membrane of the second neuron.



(a) (b)

Figure 16: Results of the delay measurments on the hardware. (a) Delay depending on the amount of repeaters needed for the connection. (b) Exemplary look of the delay measurment for a single spike of the presynaptic neuron. The black line gives the spiketime of the presynaptic neuron, the red line the extracted arrival of the signal at the postsynaptic neuron

The peakstart is defined as the point, when the membrane potential of the postsynaptic neuron first starts to rise. A brief overview of the algorithm used to determine it is given in the following. Starting at the known spiketime of the presynaptic neuron, a window with fixed size is shifted through the data points of the membrane potential. This is done up to 10 ms after the initial spike has occurred, as the signal is certain to arrive within this interval. The peakstart is then defined as the start of the interval that has the largest standard deviation. This approach works because the synaptic kernel of the hardware is an exponential kernel and thus the slope is strongest at the start of the signal. For the measurements in Fig. 16, the window length is chosen to be 20 data points, which translates to roughly 2 ms.

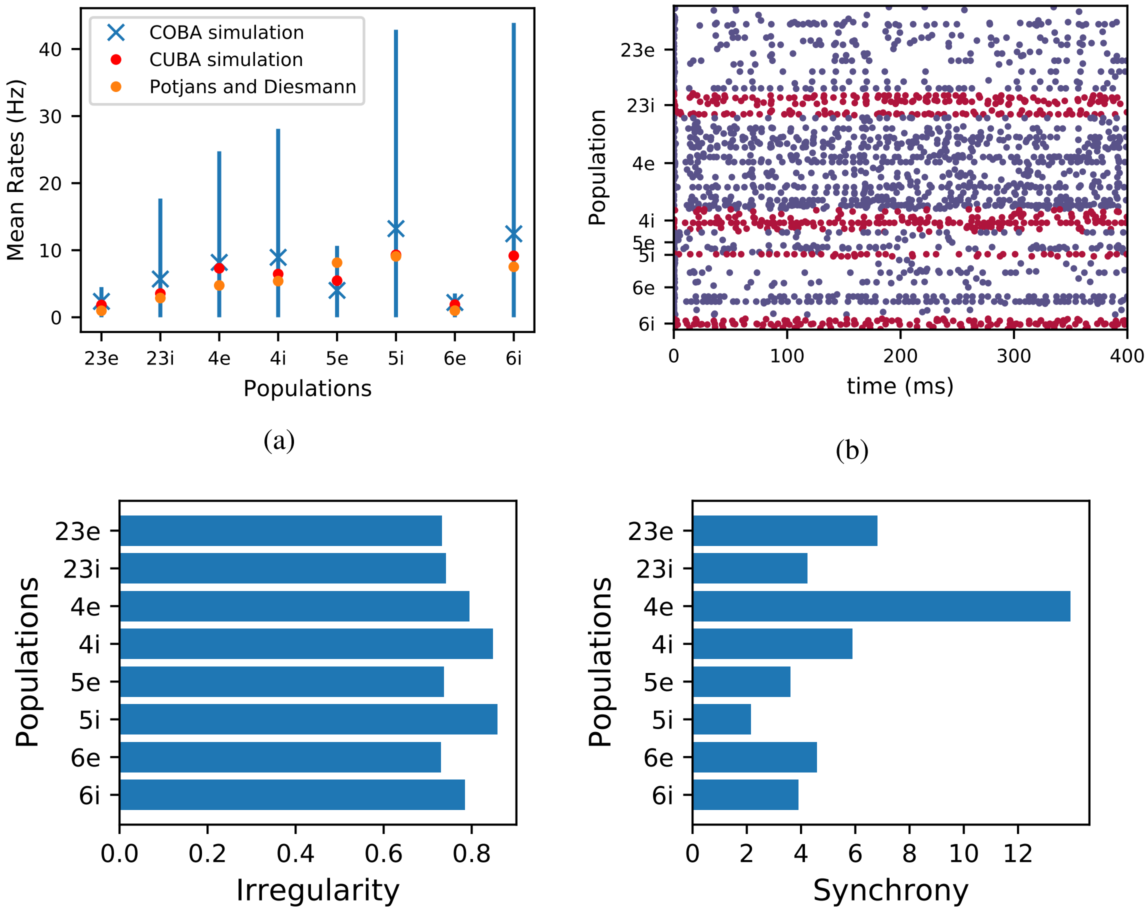
After loading the calibration process, that minimizes the fixed pattern noise explained in section 3.2, the described network is run with a simulation time of 1200 ms bio time, and the parameters of the first neuron are set so that it spikes roughly once every 100 ms for a total of 12 spikes within the emulated time span. Fig. 16 (a) shows the ensuing delay values depending on the amount of HICANNs that are between both neurons. They are averaged over all of the spikes that happen within a single run. An exemplary look of the calculation of a single delay value is given in Fig. 16 (b). For every HICANN that is passed, there is a repeater to keep the signal strength the same. With regard to this, an increased delay for a further connection is expected. Depending on the placement of the postsynaptic neuron, the measured delays are in between 0.2 ms for a measurement on the same HICANN and 1.3 ms for a placement with 18 repeaters in between. These values only represent a minimal estimate for the delays expected on the wafer, since it is not guaranteed that the synapse is already at full strength when the first signal is observed at the postsynaptic neuron. Additionally, it was not possible to realise a connection over the maximal distance of the wafer, as the connection becomes increasingly susceptible to hardware specific errors. Bearing this in mind, the delay is expected to be larger than what is shown in Fig 16 (a). To be on the safe side and in accordance with previous studies (Kaiser, 2020; Petrovici et al., 2014), the delay distribution is assumed to be centered around 1.5 ms with a spread of 50%.

3.4 Results for Fully Adapted Model

This section presents a simulation that predicts the behaviour of the cortical column under the restrictions that will eventually be forced upon the model when it is run on the BSS-1 hardware.

Effectively, this means combining the aspects that have been simulated independently before, namely the parameter variation from section 3.2 with the realistic delay values found in section 3.3.3. They are added to the scaled COBA LIF simulation, with the results of this simulation being shown in Fig. 17.

The mean rates are within a range of at most 5 Hz to both, the rates from the scaled CUBA model and the rates from the initial publication by Potjans and Diesmann. The increase that is seen for most of the populations has already been observed and discussed in section 3.2. There also is an increased spread of the firing rates within each population. This effect is already noticeable when applying the parameter variation, but has not been discussed as of yet. A large part of this effect can be traced to the variation of the threshold potential. Since the lower boundary of the distribution is directly at the reset potential, there are now some neurons with a threshold that is only slightly above the reset potential. These will fire with a rate close to the maximum firing rate that is given through the refractory period: $1/\tau_{\text{ref}}$, widening the distribution considerably. The raster plot does not show any signs of synchronous behaviour, a fact that is backed up by the irregularity and synchrony values that have stabilized when compared to the delay investigation. Irregularities are centered around 0.8 with almost no deviation between the different populations and synchrony values only show the same increase for population 4e that has been seen in section 3.1 and section 3.2.



(a)

(b)

(c)

(d)

Figure 17: Results for the final version of the COBA model, combining the parameter variation with the delay estimate on the hardware. (a) Mean firing rates, compared to the initial values from Potjans and Diesmann and the values of the 10% CUBA simulation that was the starting point of this thesis. (b) Raster plot showcasing the asynchronous irregular firing regime of the final model. (c) Irregularity values and (d) synchrony values.

Even when taking a look at the worst case scenario for the synaptic delay, the observed firing behaviour of the model still displays an asynchronous irregular pattern with a maximum of 5 Hz as deviation of the mean firing rates. The instabilities seen within section 3.3 do not appear. The parameter variation between individual neurons seems to shut down any synchronous behaviour the system might exhibit. These results also hold for differently seeded networks, confirming that it is not just a single convenient setup. Even the adjustments to the reversal potentials, that were needed in section 3.3.1, are now reversible without the loss of the asynchronous irregular firing pattern. These observations are considered sufficient to conclude that the model is able to withstand a hardware implementation, without a drastic change of its behaviour.

4 Discussion

This thesis deals with the feasibility of implementing the model of a cortical column, a structure that is found in the cerebral cortex of mammals, on the neuromorphic hardware BrainScaleS-1 of the Electronic Vision(s) group in Heidelberg. To that end, a scaled version of the initial model (Potjans & Diesmann, 2012), run with the neural network simulator NEST (Gewaltig & Diesmann, 2007), is gradually adapted to account for most of the limitations of the system. The established model serves as a first indication of what is expected for a possible hardware implementation.

The first adjustment is to the underlying neuronmodel. It was changed from a CUBA LIF model to the COBA LIF model that is employed on the hardware. The chosen approach tries to keep the same synaptic input in both scenarios. In spite of the noticeable difference between the dynamics of both models, even without further tuning, only a slight deviation of the firing rates is observed. Considering irregularity and synchrony of the network, the behaviour of the neurons can still be classified as aynchronous irregular. In comparison to the other effects that are investigated, the change of the neuronmodel is the only one that causes an explicit change in the equations describing the dynamics of the model, so being able to reproduce the firing pattern of the initial column with COBA synapses is already quite promising for the future hardware implementation.

The next step is taking into account the fixed pattern noise of the hardware. Due to the manufacturing process of the chip, small deviations between the parameters of individual neurons are unavoidable. The effects are estimated through a normal distribution, a choice motivated in Schwarzenböck (2019). Implementing these changes to the COBA simulation leads to a slight increase of the firing rates and a wider spread within the rates of each population. With a maximum increase of roughly 1 Hz for excitatory and 3 Hz for inhibitory populations, it is not expected to cause major problems to the hardware implementation.

As a last adaption to the model, the effects that are induced by a different synaptic delay, the time it takes for the signal to propagate from sending to receiving neuron, are examined. From two milliseconds onward, an increase of the mean delay value results in a synchronization of the firing rates. The exact point at which this transition occurs depends on the width of the distribution. Additionally, this effect can be compensated for through the use of larger reversal potentials. To obtain an estimate over the firing regime that is expected on BSS-1, the synaptic delay on the hardware is evaluated, using a simple network of only two neurons. This small experiment leads to the conclusion that a delay distribution between 0.5 ms and 2 ms is realistic for the hardware. These values are far below the critical point observed in the software analysis, proving that the delays on the hardware are not a hindrance for the cortical column model.

In the end, all the adjustments are combined into one simulation that serves as a means of predicting the behaviour of the column for the hardware implementation. Despite the amount of adjustments that have been performed, the simulation still shows the asynchronous irregular firing pattern of the original model, with only a small deviation of the mean firing rates. It is to be noted, that the discussed results are not definitive. There are still effects that have not yet been accounted for, such as for example the synapse loss on the hardware. Furthermore, even though the variation used for the neurons was based on previous experiments, there are no explicit measurements to confirm them. For the synaptic time constant in particular, the mean value is not well known and a standard deviation of 20 % might not be enough to predict the whole influence it has on the network.

Despite these considerations, the results of this thesis take into account the main problems that occur, when considering a hardware implementation of the cortical column. The established software simulation includes the main changes, that are needed for the model to be successfully run on the hardware. It demonstrates, that the model of a cortical column is able to withstand the imperfection of an analogue computing substrate and provides data that can be used to critically asses the results of a future hardware implementation.

5 Outlook

In its current state, the software simulation developed in this work already includes most of the aspects that play into the final hardware implementation of the cortical column. Nevertheless, there are still some features that differ between the numerical simulation and the expected emulation on BrainScaleS-1.

One such aspect is the loss of individual connections, called synapse loss, that is caused by the limited amount of resources that are available on a chip. During the experiments to determine the delay values on BSS-1, it proved to be a common occurrence when connecting HICANNs on far ends of the wafer. Even though a small amount of lost synapses should not affect a network of this scale overly much, a software implementation might still be helpful to confirm this assumption. It can then also help with a possible compensation for lost synapses through the adjustments of other synaptic weights. Apart from that, it is questionable how much more insight can be gained through further optimization of the software simulation, since in the end, it is only a numerical simulation that cannot predict all of the eventualities that might occur because of the analogue nature of BSS-1. Furthermore, with hardware in the loop training that is available on the analogue system, it might even be possible to minimize the predicted deviations between the CUBA and COBA model.

Finally, a successful implementation would serve as a promotion for both the model and the hardware. On the one hand, it would proof the viability of the model that is able to function, even when faced with the imperfections found in analogue substrates like BSS-1 and the human brain. On the other hand, it would serve as a benchmark for the BrainScaleS-1 system and show that analogue neuromorphic hardware is very well suited for future investigations of the human brain.

References

Albada, S. J., Rowley, A. G., Senk, J., Hopkins, M., Schmidt, M., Stokes, A. B., Lester, D. R., Diesmann, M., & Furber, S. B. (2018). Performance comparison of the digital neuromorphic hardware spinnaker and the neural network simulation software nest for a full-scale cortical microcircuit model. *Frontiers in Neuroscience*, *12*, 291. https://doi.org/10.3389/fnins.2018.00291

Albada, S. J., Helias, M., & Diesmann, M. (2015). Scalability of asynchronous networks is limited by one-to-one mapping between effective connectivity and correlations. *PLOS Computational Biology*, *11*(9), 1–37. https://doi.org/10.1371/journal.pcbi.1004490

Beul, S. F., & Hilgetag, C. C. (2015). Towards a “canonical” agranular cortical microcircuit. *Frontiers in Neuroanatomy*, *8*, 165. https://doi.org/10.3389/fnana.2014.00165

Brunel, N. (2000). Dynamics of sparsely connected networks of excitatory and inhibitory spiking neurons. *Journal of Computational Neuroscience*, *8*. https://doi.org/10.1023/A:1008925309027

Cain, N., Iyer, R., Koch, C., & Mihalas, S. (2016). The computational properties of a simplified cortical column model. *PLOS Computational Biology*, *12*(9), 1–18. https://doi.org/10.1371/journal.pcbi.1005045

Chellappa, R., Sinha, P., & Phillips, P. J. (2010). Face recognition by computers and humans. *Computer*, *43*(2), 46–55. https://doi.org/10.1109/MC.2010.37

Davison, A., Brüderle, D., Eppler, J., Kremkow, J., Muller, E., Pecevski, D., Perrinet, L., & Yger, P. (2009). Pynn: A common interface for neuronal network simulators. *Front. Neuroinform.* https://doi.org/10.3389/neuro.11.011.2008

Eilers, J. (2019). Nervenzellen. In R. Brandes, F. Lang, & R. F. Schmidt (Eds.), *Physiologie des menschen: Mit pathophysiologie* (pp. 57–64). Berlin, Heidelberg, Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-662-56468-4_5

Fakler, B., & Eilers, J. (2019). Ruhemembranpotenzial und aktionspotenzial. In R. Brandes, F. Lang, & R. F. Schmidt (Eds.), *Physiologie des menschen: Mit pathophysiologie* (pp. 65–71). Berlin, Heidelberg, Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-662-56468-4_6

Gerstner, W., Kistler, W., Naud, R., & Paninski, L. (2014). *Neuronal dynamics*. Cambridge University Press.

Gewaltig, M.-O., & Diesmann, M. (2007). Nest (neural simulation tool). *Scholarpedia*, *2*(4), 1430.

Grübel, A., & Baumbach, A. (2017). *F09/f10 neuromorphic computing*. https://www.physi.uni-heidelberg.de/Einrichtungen/FP/anleitungen/F09.pdf

Hbp neuromorphic computing platform guidebook. (n.d.). https://electronicvisions.github.io/hbp-sp9-guidebook/pm/pm_hardware_configuration.html#the-neuromorphic-wafer-module (accessed: 14.11.2020)

Jordan, J., Ippen, T., Helias, M., Kitayama, I., Sato, M., Igarashi, J., Diesmann, M., & Kunkel, S. (2018). Extremely scalable spiking neuronal network simulation code: From laptops to exascale computers. *Frontiers in Neuroinformatics*, *12*, 2. https://doi.org/10.3389/fninf.2018.00002

Kaiser, J. (2020). *Implementation of large scale neural networks on the neuromorphic brainscales-1 system* (Masterarbeit). Universität Heidelberg.

Müller, E., Schmitt, S., Mauch, C., Billaudelle, S., Grübl, A., Güttler, M., Husmann, D., Ilmberger, J., Jeltsch, S., Kaiser, J., Klähn, J., Kleider, M., Koke, C., Montes, J., Müller, P., Partzsch, J., Passenberg, F., Schmidt, H., Vogginger, B., . . . Schemmel, J. (2020). The operating system of the neuromorphic brainscales-1 system.

Petrovici, M. A. (2016). *Form versus function: Theory and models for neuronal substrates*. Springer.

Petrovici, M. A., Vogginger, B., Müller, P., Breitwieser, O., Lundqvist, M., Muller, L., Ehrlich, M., Destexhe, A., Lansner, A., Schüffny, R., Schemmel, J., & Meier, K. (2014). Characterization and compensation of network-level anomalies in mixed-signal neuromorphic modeling platforms. *PLOS ONE*, *9*(10), 1–30. https://doi.org/10.1371/journal.pone.0108590

Plesser, H., Eppler, J., Morrison, A., Diesmann, M., & Gewaltig, M.-O. (2007). Efficient parallel simulation of large-scale neuronal networks on clusters of multiprocessor computers. https://doi.org/10.1007/978-3-540-74466-5_71

Potjans, T. C., & Diesmann, M. (2012). The Cell-Type Specific Cortical Microcircuit: Relating Structure and Activity in a Full-Scale Spiking Network Model. *Cerebral Cortex*, *24*(3), https://academic.oup.com/cercor/article-pdf/24/3/785/14099777/bhs358.pdf, 785–806. https://doi.org/10.1093/cercor/bhs358

Rhodes, O., Peres, L., Rowley, A. G. D., Gait, A., Plana, L. A., Brenninkmeijer, C. Y. A., & Furber, S. B. (2019). Real-time cortical simulation on neuromorphic hardware. *CoRR*, *abs/1909.08665*arXiv 1909.08665. http://arxiv.org/abs/1909.08665

Schemmel, J., Brüderle, D., Grübl, A., Hock, M., Meier, K., & Millner, S. (2010). A wafer-scale neuromorphic hardware system for large-scale neural modeling. *Proceedings of the 2010 IEEE International Symposium on Circuits and Systems (ISCAS"10)*, 1947–1950.

Schwarzenböck, Q. (2019). *Towards balanced random networks on the brainscales i system* (Bachelor). Universität Heidelberg.

Thakur, C. S., Molin, J. L., Cauwenberghs, G., Indiveri, G., Kumar, K., Qiao, N., Schemmel, J., Wang, R., Chicca, E., Olson Hasler, J., Seo, J.-s., Yu, S., Cao, Y., van Schaik, A., & Etienne-Cummings, R. (2018). Large-scale neuromorphic spiking array processors: A quest to mimic the brain. *Frontiers in Neuroscience*, *12*, 891. https://doi.org/10.3389/fnins.2018.00891

Wagatsuma, N., Potjans, T., Diesmann, M., & Fukai, T. (2011). Layer-dependent attentional processing by top-down signals in a visual cortical microcircuit model. *Frontiers in Computational Neuroscience*, *5*, 31. https://doi.org/10.3389/fncom.2011.00031

Weidner, J. (2019). *Experiment visualization and simulations towards a cortical microcircuit on the brainscales neuromorphic hardware* (Bachelorarbeit). Universität Heidelberg.

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, den 16.11.2020,

Moritz Hornung