Department of Physics and Astronomy

University of Heidelberg

Bachelor Thesis in Physics

submitted by

Laura Kriener

born in Offenburg (Germany)

September 2014

Binaural Sound Localization on Neuromorphic Hardware

This Bachelor Thesis has been carried out by
Laura Kriener
at the
Kirchhoff-Institute for Physics
Ruprecht-Karls-Universität Heidelberg
under the supervision of
Prof. Dr. Karlheinz Meier

Abstract

In this thesis binaural sound localization is performed on neuromorphic hardware. The neural network described in the Jeffress model locates the azimuthal position of a sound by analysing the interaural time difference, so called ITD, of the sound arriving at both ears. Due to the periodicity of pure tones, the response of the network is ambiguous. By combining multiple networks for different spectral components of the sound, the Jeffress model is extended to a network that yields an unambiguous answer. In preceding studies by the author the functionality of this combination of networks has been shown in software simulations.

Here this network is implemented on the neuromorphic microchip Spikey. Methods to overcome the obstacles imposed by the limited signal bandwidth of the chip as well as inhomogeneities in the hardware components are developed. In this context a formerly unknown effect of interaction between input signals, which impairs ITD-detection, was measured and investigated in detail. The knowledge obtained by these measurements allows a modification of the network to decrease the impact of the signals' interaction. With the modified network a successful ITD-detection is performed.

Zusammenfassung

In dieser Arbeit wird binaurale Schalllokalisierung auf neuromorpher Hardware gezeigt. Das Jeffress-Modell beschreibt ein neuronales Netzwerk, welches die azimutale Position eines Geräuschs lokalisiert, indem es die interauralen Zeitdifferenzen, so genannte ITDs, analysiert. Aufgrund der Periodizität von reinen Tönen ist das Ergebnis des Netzwerks mehrdeutig. Durch die Kombination mehrerer Netzwerke, die auf unterschiedlichen Frequenzen des Schalls arbeiten, kann das Jeffress-Modell zu einem Netzwerk erweitert werdem, das eindeutige Ergebnisse liefert. Vorangegange Arbeiten der Autorin haben bereits die Funktionalität dieses kombinierten Netzwerks in Software-Simulationen gezeigt.

In dieser Arbeit wird das kombinierte Netzwerk auf den neuromorphen Mikrochip Spikey übertragen. Dazu werden Methoden zum Ausgleich von Inhomogenitäten in der Hardware sowie zur Kompensation der limitierten Bandbreite des Chips entwickelt. Dabei wurde eine bis dahin unbekannte Interaktion zwischen Eingangssignalen, welche die Detektion von ITDs erschwert, beobachtet. Erkenntnisse, die bei der detaillierten Untersuchungen dieses Effekts gewonnen wurden, erlaubten eine Modifikation des Netzwerk, welche die Interaktion der Eingangssignale reduziert. Mit diesem modifizierten Netzwerk konnten erfolgreich ITDs detektiert werden.

Contents

1 Introduction

The immense amount of funding received by recent projects like the European Human Brain Project (2014) or the American BRAIN-Initiative (2014) shows that understanding the brain has become a topic of great interest in modern science. Both projects combine experimental and computational neuroscience to gain new insights into the functional principles of the brain. One aim of computational neuroscience is to understand the mechanisms in the brain by simulating the behaviour of neural networks. But the simulation of large networks on conventional computers is very demanding concerning computational power and simulation time. In contrast, the rather recently developed technology of neuromorphic hardware models the components of the brain with electrical circuits mimicking the electrical behaviour of neurons and synapses. These physical models evolve in parallel and in continuous time. The hardware is therefore able to perform massively parallel computations similar to the brain. In the context of the FACETS (2010) and BrainScaleS (2014) research projects, the neuromorphic microchip Spikey was developed.

A possible application of neuromorphic hardware is the emulation of biologically inspired networks. Previous works at the Kirchhoff-Institute for Physics in Heidelberg investigated a model that autonomously learns to improve phase-locking as e.g. observed in the auditory system of barn owls (Gerstner et al., 1996). In this context, it was shown that phase-locking can be performed on the Spikey chip (Scherzer, 2013; Pfeil et al., 2013b). Since phase-locked spike trains are essential for binaural sound localization, the next step is to implement and investigate the networks responsible for the actual localisation of a sound. In 1948, Lloyd Jeffress introduced the model of a network capable of locating the azimuthal position of a sound source by determining the time difference between the sound at the left and right ear (Jeffress, 1948). Biological evidence for the model has been found more than 40 years later (Carr and Konishi, 1990).

This thesis aims to implement the Jeffress model on the Spikey hardware system. Additionally, the network is extended to overcome the ambiguity of the answers obtained by the Jeffress model. The ambiguity is caused by periodic input and can be avoided by combining multiple networks that use different frequencies. In the context of preparative studies the functionality of the combined network as well as its dependence on network and input parameters were investigated in software simulations (Kriener, 2014). The focus of this work lies on overcoming obstacles imposed by the limited neuron and synapse count on the chip, the narrow parameter range of neuron and synapse parameters as well as the influence of inhomogeneities in the electric components. First, a network similar to the one tested in simulation is implemented on hardware and calibration algorithms to compensate for inhomogeneities and imperfections in hardware are developed. In this context, a formerly unknown effect of interaction between input signals was found and investigated in detail in a second step. Finally, with the knowledge obtained by the previous steps, a modified network is introduced and its functionality is tested qualitatively.

1.1 Biological Background

The Jeffress model (Jeffress, 1948) proposes a network for binaural sound localization, which locates a sound by analysing the *interaural time difference* (ITD) of the sound signal. If the source of the sound is positioned on either side of the head, the sound waves travel different distances to the ear which is closer to the source of the sound and the one which is farther away. Therefore, the sound waves arrive at the ears with a different phase. For a pure tone of frequency f and amplitude a_0, the sound waves $a(t)$ and $a'(t)$ arriving at the ears can be described as

$$a(t) = a_0 \cdot \sin(2\pi f t), \qquad \text{(left ear)}$$
$$a'(t) = a_0 \cdot \sin(2\pi f(t + \Delta t)), \qquad \text{(right ear)}$$

where Δt describes the ITD.

Before being interpreted by a neural network, the cochlea transforms the sound waves into spike coded information. In that process, the frequency components of the sound are split up into spike trains on different auditory nerve fibres. The resulting spike trains are phase-locked. This means that an action potential is triggered with the highest probability at a certain phase of the incoming tone (Gerstner and Kistler, 2002).

As input from the left and right ear the Jeffress model assumes two regular spike trains which are shifted in time by Δt. The temporal distance between two spikes in these spike trains is given by the cycle duration of the incoming sound wave. Figure 1 shows the network architecture of an *ITD-detector* as described in the Jeffress model. Each neuron is connected to the auditory nerves of both ears. The connecting fibres delay the incoming spike trains of the left and right ear. Since the fibres vary in length, each neuron receives its input signals with additional delays, which results in an characteristic difference between the left and right delay for each neuron. By assigning each neuron to an individual difference between the left and right delay (in the following called *characteristic delay*) the ITD-detector generates a spatial coding of ITDs. For each pair of signals, one of the characteristic delays compensates the ITD and therefore the corresponding neuron receives the input spikes from the left and right ear coincidently. Then, this neuron fires with an higher rate than the other neurons.

Because of the periodicity of the spike trains, the network described above is only able to detect phase differences between two input signals rather than the absolute time difference. For two pure tones with the cycle duration T and the ITD Δt all pairs of pure tones with an interaural time differences $\Delta t'$ of the form

$$\Delta t' = \Delta t + n \cdot T \qquad n \in \mathbb{Z}$$

result in the same spike trains. Therefore, not only the neuron whose delay compensates Δt ($n = 0$), but also neurons that correspond to delays compensating $\Delta t'$ with $n \neq 0$ fire with the highest rate. The network's response is ambiguous and is only able to detect the phase difference between two signals. Different ITDs causing the same phase difference of the input signals can not be distinguished.

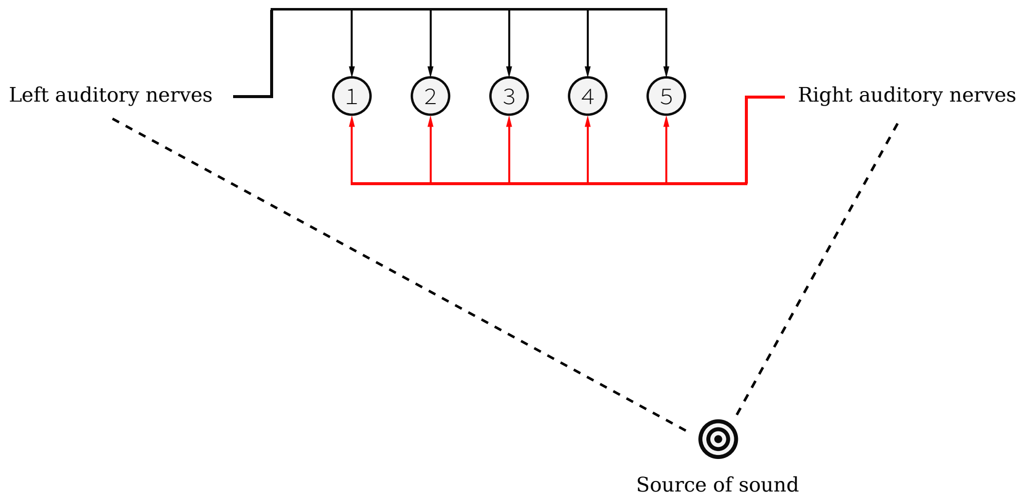

Figure 1: Jeffress model with a source of sound on the right side of the head. The neurons (circles) receive their input from left and right ear by the auditory nerve fibres. These fibres, also called *delay lines*, vary in length for the different neurons. Because of that they add characteristic delays to the incoming signals. In this example the sound waves arrive earlier at the right ear than at the left. The delays added by the delay lines of neuron number 1 compensate the ITD of the incoming signal. This neuron therefore fires with the highest rate.

The $\Delta t'$ causing an ambiguous response of the network depend on the cycle duration of the input signals. When using multiple detectors for different input frequencies all of them show ambiguous responses. However, the neurons firing with the highest rate because of the delays corresponding to $\Delta t'$ with $n \neq 0$ are most likely different for each detector. Only the neuron corresponding to the real interaural time difference Δt is the same in all detectors. This allows to detect absolute time differences using a network combining multiple ITD-detectors.

1.2 Preparative Studies

During an internship preceding this thesis (Kriener, 2014), the described networks were investigated with software simulations using PyNN (Davison et al., 2008) with NEST (Gewaltig and Diesmann, 2007) as a backend. In the following, the results important for an implementation of the network on neuromorphic hardware are presented. All results can be found in detail in Kriener (2014).

Neurons in an ITD-detector should fire with a higher rate if their input spike trains coincide. Simulations have shown that this behaviour only occurs for a small range of synaptic weights connecting the input to the neurons. If the synaptic weight is too low, the neuron will not spike at all. If the weight is too high, the neuron will spike with every incoming spike and its firing rate is therefore not dependent on the ITD of its input. Even within the described range the dependency of the firing rate on the synaptic weight is very strong. This suggests that on hardware a careful calibration of the synaptic weights is necessary.

Simulations showed that 3 ITD-detectors for different frequencies are enough to resolve the ambiguous responses of the detectors caused by the periodic input. But the frequencies have to be chosen carefully, since input frequencies that are multiples of each still lead to ambiguous responses of the network.

There are two possible ways for extracting the network's response to a certain input. The firing rates of the neurons corresponding to the same delay in each detector can be summed up, which results in a distribution of firing rates over the characteristic delays. Ideally this distribution has one clear maximum for the delay that compensates the ITD and other lower maxima caused by the delays compensating $\Delta t'$ with $n \neq 0$. By finding the maximum of the distribution the delay compensating the ITD, and by that the ITD itself, can be determined.

To increase the difference between the maximum and the background an *integration layer* can be used. It consists of an additional population of neurons of the same size as a detector. All neurons of the detectors corresponding to the same delay are connected excitatory to one of the neurons in the integration layer. As the output firing rate of the neuron rises more than linear with the input rate, the result of the network will be modulated compared to the summing up of firing rates. High input rates are amplified, whereas low rates are damped. Therefore, the difference between maximum and background is increased.

The frequency components of the sound signal are assumed to be equally important and therefore the maximum firing rates of the detectors for that frequencies should be approximately equal. Otherwise, the result of one detector might dominate the overall result. Forcing neurons with different input frequencies to fire with the same maximum rate requires an even more careful choice of the synaptic weights than when dealing with only one ITD-detector.

Additionally, it was found that the value for the membrane time constant of the neurons should be roughly between a half and a fourth of the cycle duration of the input frequencies. A lower the time constant leads to a better time resolution. Since the configurable range for the membrane time constants on hardware is roughly between $3\,\mathrm{ms}$ and $10\,\mathrm{ms}$ (biological time domain), the input frequencies should be chosen around $100\,\mathrm{Hz}$.

1.3 Spikey

The neuromorphic hardware system used in this work is the $5 \times 5\,\mathrm{mm}^2$ microchip *Spikey* developed in context of the research project FACETS (2010). It contains 384 circuits modelling the electrical behaviour of neurons. Compared to biology the hardware is up to 10^5 times accelerated although in this study a speed-up of 10^4 is used. If not stated otherwise, all times and frequencies mentioned in the following are given in the biological time domain.

A detailed compilation of the used hardware as well as the used software stack can be found in table 2 in appendix A.

1.3.1 Neuron and Synapse Model

The implemented neuron model is the leaky integrate-and-fire neuron model with conductance based synapses (Dayan and Abbott, 2001).The dynamics are given by

$$C_\mathrm{m}\frac{\mathrm{d}V_\mathrm{m}}{\mathrm{d}t} = -g_\mathrm{l}(V_\mathrm{m} - E_\mathrm{l}) - \sum_i g_i(V_\mathrm{m} - E_i).$$

C_m describes the neuron's membrane capacitance, V_m the membrane potential and g_l the leakage conductance. If there is no synaptic input $\tau_\mathrm{m} = \frac{C_\mathrm{m}}{g_\mathrm{l}}$ is the time constant of the decay of V_m towards the leakage reversal potential E_l. The sum over all synapses i of the neuron adds up the synaptic input. The synaptic input is determined by a synaptic conductance g_i that drives the membrane potential towards the synapse's reversal potential E_i. For excitatory synapses the reversal potential is given by $E_i = E_\mathrm{exc}$ and for inhibitory by $E_i = E_\mathrm{inh}$. The time course of the synaptic conductance g_i is described by

$$g_i(t) = p_i(t) \cdot w_i \cdot g_i^\mathrm{max},$$

where $p_i(t)$ approximately represents an exponential decay, w_i the synaptic weight and g_i^max the maximum value of the conductance. For a more detailed description see Brüderle (2009) and Pfeil et al. (2013a).

1.3.2 Hardware Specifications

Spikey is divided into two blocks, each containing 192 neurons and 256 synapse line drivers. The synapse line drivers receive spike input from an external source or from neurons on the chip. They forward spike events to postsynaptic neurons to which they are connected via synapses. The strength of these connections is regulated by the synaptic weight w. Since every synapse line driver can be connected to every postsynaptic neuron in the same neuron block there are $256 \cdot 192 = 49152$ synapses. Due to technical reasons only the second neuron block is accessible. Figure 2 shows a micro photograph of the chip as well as a schematic drawing of the signal transmission between synapse line driver and neuron.

The synapse line driver receives a pulse event and transforms it into a linear voltage ramp. The time course and the parameters influencing the voltage ramp are shown in figure 3. The rising slew rate, controlled by I_rise, is very high, whereas the falling slew rate I_fall is much smaller and corresponds to the synaptic time constant τ_syn. The height of the voltage ramp is determined by I_out controlling g_i^max. The synapses transform the voltage ramp into a current with an exponential time course also shown in figure 3 and scale this current with the synaptic weight w_i. At the neuron the current is transformed into the time dependent conductance $g_i(t)$. For details see Schemmel et al. (2007) and Brüderle (2009).

Some of the described parameters of neurons, synapses and synapse line drivers can be configured individually for each instance but others are shared parameters. Table 1 shows the parameters, important for this work as well as their configurability.

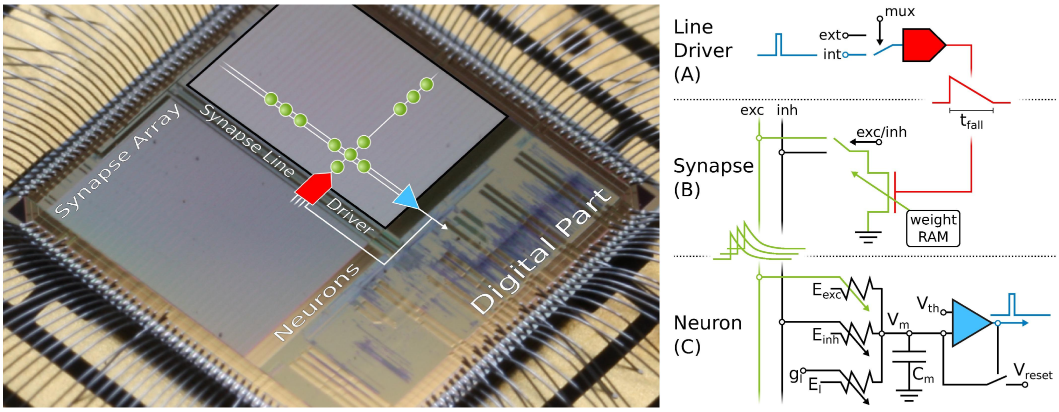

Figure 2: The photography on the left shows the Spikey chip and its division into two neuron blocks as well as the location of synapse line drivers, synapses and neurons. The right side is a schematic drawing of the signal transmission between synapse line driver and neuron. A pulse event reaches the synapse line driver and is transformed into a voltage ramp (A). In the synapse (B) the voltage ramp is converted into an exponentially decaying current which is transformed into the synaptic conductance in the neuron (C). Taken from Pfeil et al. (2013a).

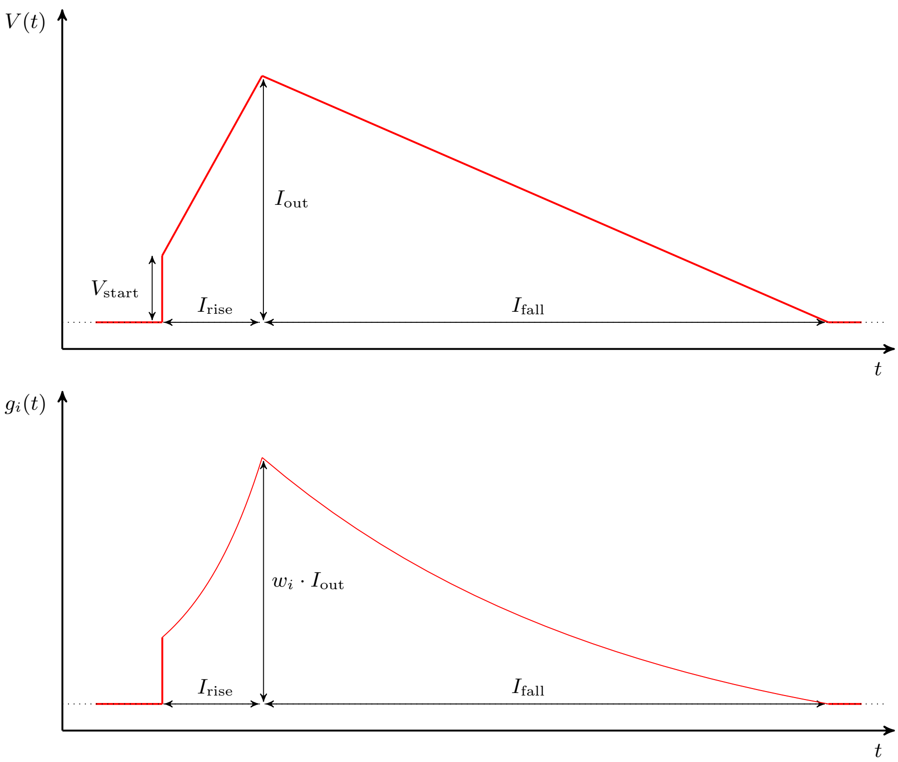

Figure 3: Top: Time course of the voltage ramp between synapse line driver and synapse. The parameters influencing its shape are depicted. Bottom: Exponential time course of the current produced by the synapse as well as of the synaptic conductance.

Scope	Name	Configurability	Description
Neuron	g_l	individual neuron	Leakage conductance
	E_l	shared odd/even neurons	Leakage potential
	E_inh	shared odd/even neurons	Inhibitory reversal potential
	E_exc	shared odd/even neurons	Excitatory reversal potential
	V_th	shared odd/even neurons	Firing threshold
	V_reset	shared odd/even neurons	Reset potential
Synapse line driver	I_rise, I_fall	individual line driver	Two bias currents for rising and falling slew rate of presynaptic voltage ramp
	g_i^max	individual line driver	Bias current controlling maximum of voltage ramp
Synapse	w	individual synapse	4 bit synaptic weight

Table 1: Parameters of neurons, synapse line drivers and synapses and their configurability.

2 Multiple Detector Network

2.1 Implementation on Spikey

Precise artificial delays are difficult to realise on hardware. Therefore, the characteristic delays of the neurons are not produced on hardware, but added in software before sending the signals into the network. It is not necessary to modify both the left and the right signal, since delaying the left signal and moving the right forward in time is equivalent. To assign the characteristic delays, the left signals are kept at the same values for all neurons and for each neuron a different delay is added to the right signal.

Figure 4 shows a schematic drawing of the synapse line driver configuration used to implement a network consisting of multiple ITD-detectors as described in section 1.1. Each neuron receives the input from the left and right ear via individual groups of synapse line drivers. Since the number of synapse line drivers is limited on the chip, it would be convenient, if synapse line drivers provided input for more than one neuron. This is not possible due to three reasons: first, the different detectors of the network receive input of different frequencies. Therefore their input spike trains are different and must come from separate synapse line drivers. Second, the neurons in one detector can not share the drivers for their right inputs, because due to the individual delays the right input spike trains are different for each neuron. Third, the left input is the same for every neuron in a detector, but due to inhomogeneities in the hardware the strength of synaptic connections between one synapse line driver and different neurons varies. This leads to a varying strength of the left input. The only parameter that could compensate these variations is the synaptic weight. But since the weight is a $4\,\mathrm{bit}$ value, its resolution may be too coarse.

To achieve a good time resolution of the network, as many neurons as possible should be used. If all of the available synapse line drivers are used for 3 ITD-detectors, a maximum number of 42 neurons per detector is possible. In this case each neuron is connected to two synapse line drivers, one for the left and one for the right input.

As described in section 1.2 an integration layer can be used to evaluate the firing rates of the detectors. Since each neuron in a detector is connected to one neuron in the integration layer via a different synapse line driver, the integration layer uses a larger proportion of the synapse line drivers than a detector. For a network consisting of 3 detectors with n neurons each, the integration layer uses $3 \cdot n$ drivers. The 3 detectors receive their input via $6 \cdot n$ drivers. If the maximum number of 256 drivers is used, a maximum number of $n = 28$ neurons per detector is possible. For a high resolution the detector network and the integration layer should be split up into different emulation runs on Spikey. In the first run the spike times of the detector network are recorded, and played back into the network of the integration layer in a second emulation run.

It is additionally possible to randomly delete spikes in the input spike trains with a fixed thinning probability p. Each spike in a spiketrain is deleted with the probability p. Simulations in Kriener (2014) have shown that for $p \leq 0.3$ this thinning of the input signals does not change the functional principle of the network, but is a possibility to continuously scale the amount of input for the network without a change of the input frequency.

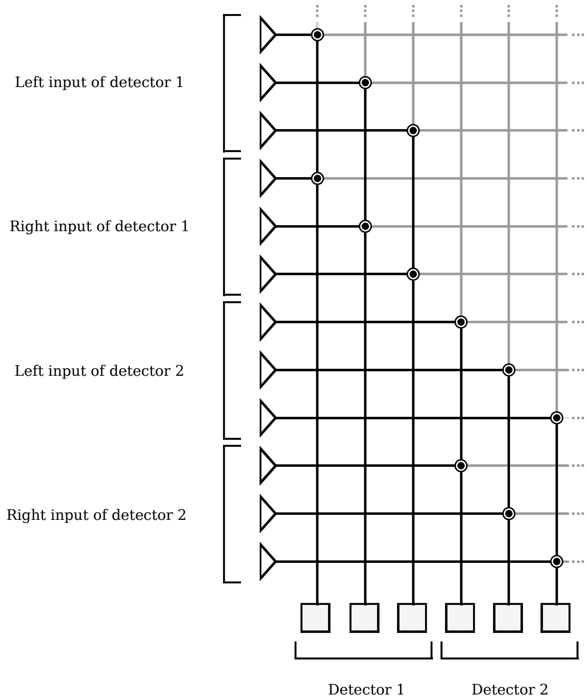

Figure 4: Schematic drawing of an exemplary network (consisting of 2 ITD-detectors with 3 neurons each) on Spikey. Each neuron (square) is connected to two synapse line drivers (triangles). The circles represent the active synapses that connect the synapse line drivers to the neurons. Via two separate synapse line drivers each neuron receives 2 input spike trains representing the spike trains from the left and right ear.

2.2 Bandwidth Limitations

The input bandwidth for digital events on Spikey depends on the distribution of the events over time as well as on their distribution over the synapse line drivers and ranges between approximately $30\,\mathrm{kHz}$ and $5\,\mathrm{kHz}$. The bandwidth is is the smallest for events occurring in very short time periods. In case of a too high input rate as many events as technically possible are sent into the system, the rest is discarded in software. This is called spike loss.

The network described above receives more than 200 spike trains with frequencies of around $100\,\mathrm{Hz}$ each. To avoid spike loss caused by the large amount of input spikes, the bandwidth should be exploited as well as possible. This can be achieved by shifting the spikes to distribute them in time. The firing rate of the neuron depends on the phase difference between its left and right input. Therefore the left and right input of each neuron must be shifted equally for each neuron as shown in figure 5. For the different neurons the spikes of each cycle duration are distributed over a certain time interval $\tilde{T}$. For $\tilde{T} = T$, where T is the cycle duration of the input, the spikes are distributed uniformly over time. In the following the *shift factor* s is defined as $s = \frac{\tilde{T}}{T}$.

Since the neurons are not interconnected with each other, the firing rate of one neuron should not depend on the timing of the inputs of all other neurons. Distributing the input

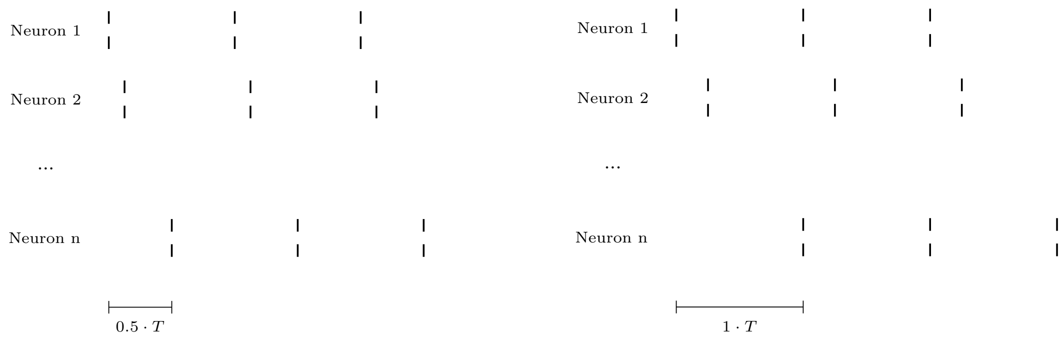

Figure 5: Shifted input spike trains (left and right input for each neuron) of a population consisting of n neurons. The shift factor ($s = 0.5$ on the left and $s = 1.0$ on the right side) corresponds to the ratio of the time interval over which the spikes of one period of duration T are distributed.

signals for the different neurons in time therefore as shown in figure 5 preserves the firing rate of the neurons as well as reduces the number of lost input spikes.

To investigate the dependency of the spike loss on the number of neurons in the network as well as on the shift factor s, additional measurements with a ITD-detector of different sizes were carried out. Figure 6 shows the dependency of the spike loss on the applied shift factor s for different numbers of neurons in the detector. Since the spike events are more evenly distributed in time for larger s, one would expect the spike loss to decrease with increasing shift factor. But instead there is an increase in the spike loss for higher s whose position depends on the number of neurons in the network.

The increase in discarded events is caused by the method of event transportation between the FPGA and the chip (Grübl, 2014). Incoming spike events are transported in packages of 3 or less events, where each of these events must be for a synapse line driver of a different one of the synapse line driver blocks. Each of these 8 blocks contains 64 synapse line drivers. With increasing shift factor s the time interval between different events increases. If the intervals grow too large the events for different synapse drivers can not be sent with the same package any more. Therefore the number of needed packages rises until its maximum is reached and spike events are dropped. The position of the second maximum depends on the number of neurons, because for larger neuron numbers more synapse line drivers are used and there are more possibilities for 3 events from different blocks to fit into one package.

This shows that for all networks the number of discarded spikes must be checked and an appropriate shift factor must be applied to the input signals.

2.3 Calibration

For a good performance of ITD detection all neurons should show the same dependence of their firing rate on the ITD. In particular, it is important that all neurons fire with the same rate for an ITD of $\Delta t = 0\,\mathrm{ms}$. Due to imperfections and inhomogeneity in the electronic components, caused by the production process, this is not automatically the case on neuromorphic hardware, even if the parameters of neurons and synapses are set

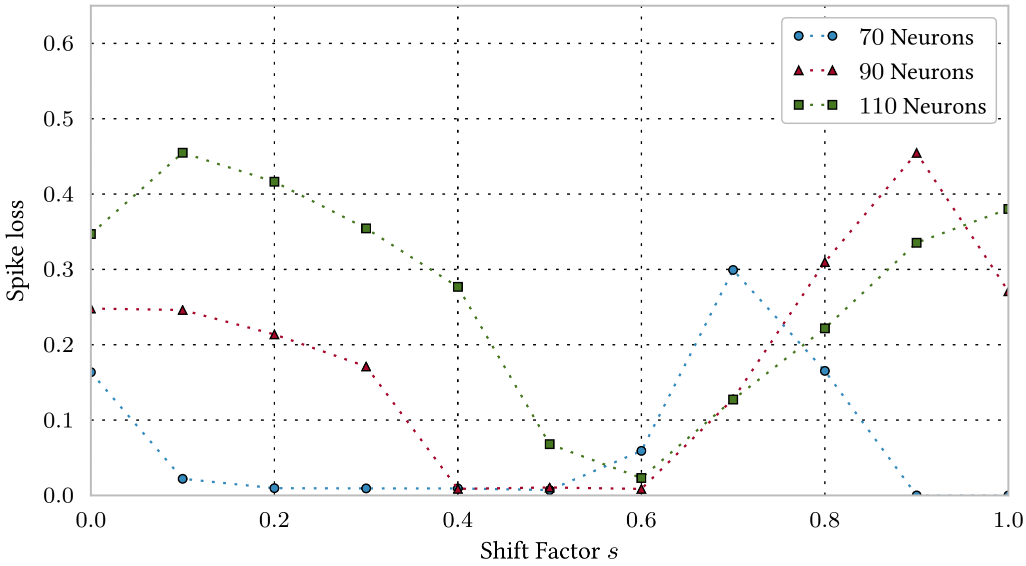

Figure 6: Ratio of discarded input spikes over shift factor s for 3 different numbers of used neurons. All neurons receive 2 input spike trains with a frequency of 100 Hz. The spike loss depends on the total number of spikes sent into the network and therefore on the number of used neurons. Shifting the input signals for the different neurons in time decreases the spike loss for small shift factors. Larger s again cause larger spike loss. This behaviour is also dependent on the number of used neurons.

to the same values. Therefore a dedicated calibration for the synapse line drivers and the leakage conductance is applied (for details about the calibration algorithms see Brüderle (2009)). A calibration of the neurons' potentials is not possible since these voltages are shared parameters.

Figure 7 shows the firing rates of four arbitrarily chosen neurons of an ITD-detector, when stimulated with two signals over the ITDs Δt between the signals. Although the synapse line driver calibration and the calibration of the leakage conductance were used and the neurons' parameters were set to the same values, the firing rates of different neurons show differences of close to 50 Hz e.g. for $\Delta t = 0$ ms. This shows that for the detection of ITDs an additional calibration is necessary. The main goal of this calibration is to compel all neurons of the different detectors, that receive inputs of different frequencies, to fire with an approximately equal maximum rate. The detection of ITDs between the left and right signal also requires the signals from the left and right ear to be equally strong. Because of that the first calibration step tries to level the strengths of the synaptic connections between the synapse line drivers and neurons. The second calibration step levels out differences between neurons caused by the combination of the left and right input by adjusting the synaptic weights.

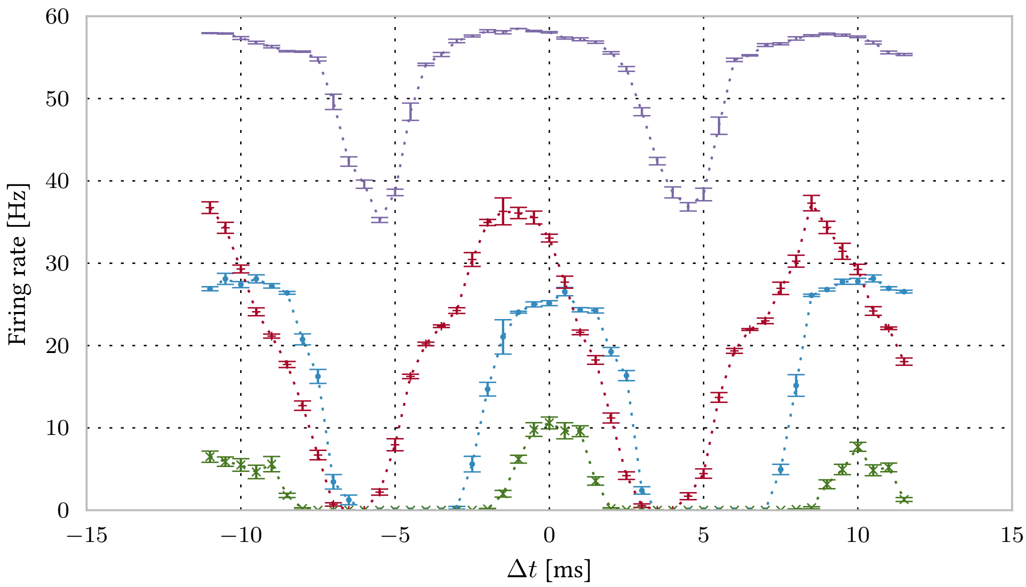

Figure 7: Firing rates of 4 different neurons (each averaged over 10 runs) with same parameter setting and same inputs plotted over ITD between left and right input. The default calibration for neuron parameters and synapse line drivers is used.

2.3.1 Calibration of Synapse Line Drivers

The synaptic weight w is the parameter that corresponds to the strength of a synaptic connection, but since it is only a 4 bit value, its resolution is very coarse. As simulation showed that the dependency on the synaptic strength is very strong, this parameter alone may not be suitable for a calibration of the synaptic connections.

The synaptic strength is also influenced by the hardware currents I_{fall} and I_{out} controlling the voltage ramp between the synapse line driver and the synapses. They can be configured for each synapse line driver individually but since the neurons in this network do not share synapse line drivers, these parameters are not shared between active synapses and can be used to adjust a synaptic connection. The resolution of these parameters is higher than the resolution of the synaptic weight. Figure 8 shows the dependency of the synaptic strength, measured by the height of an excitatory post synaptic potential (PSP), on the parameters I_{fall} and I_{out} for an exemplary synapse line driver and neuron. Although the functional correlation is similar for all combination of driver and neuron, the absolute values differ. It is therefore difficult to predict a suitable combination of I_{fall} and I_{out} for a specific pair of neuron and synapse line driver.

The following calibration algorithm is designed for a network consisting of neurons that are connected to 2 synapse line drivers. They correspond to the signals from the left and right ear and are in the following called the left and right driver of the neuron:

In a calibration of a network to very precise firing rates, hardware effects like leakage currents through synapses have to be considered. They are dependent on the used network topology, input signals and the configuration of unused synapse line drivers. Their

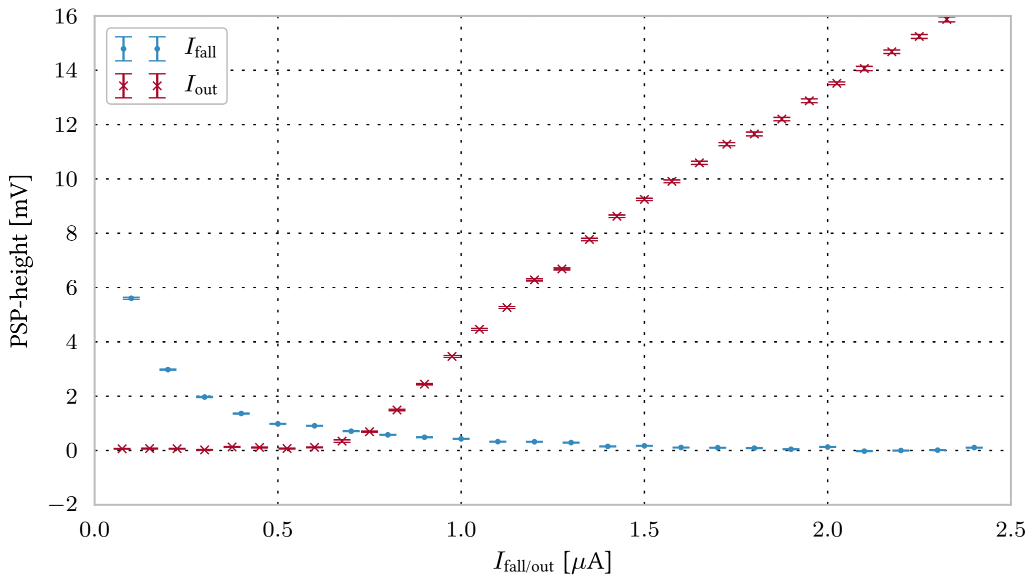

Figure 8: Dependency of the PSP-height (measure for strength of synaptic connection) on the calibration parameters I_{out} and I_{fall} for an arbitrarily chosen combination of neuron and synapse line driver. The PSP-height increases approximately linearly with I_{out} and decreases strongly with I_{fall}. Each data point was averaged over 20 emulation runs.

influence on the firing rates of the network are hard to predict. It is therefore necessary to compensate them by calibration. This can only be achieved if the network topology used in calibration is as similar as possible to the one used in later experiments. Since the synapse line drivers in the later network are used in parallel, a calibration algorithm that calibrates them in parallel is required.

This calibration algorithm consists of an iterative combination of bisection methods. To ensure that the inputs from left and right side are equally strong, each side is calibrated separately while the synaptic weight of the other side is set to zero. In the network used in experiments each neuron receives input from two synapse line drives. If, during calibration, one of these drivers is connected to the neuron with the synaptic weight zero, the weight for the connection of the other driver must be doubled to achieve the same strength of the input for the neuron as in later experiment. It is also necessary to repeat the calibrations of left and right side iteratively, since the firing rates of the neurons are dependent on the configuration of the unused drivers, in this case the drivers of the side currently set to weight zero.

For the calibration of the right side the parameters I_{fall} and I_{out} of the left drivers are set to the values obtained by the default calibration. Then the firing rates of each neuron, when presented with a signal of the same frequency as the one that is used in the later experiments, are measured and compared to a configurable target firing rate. The target firing rate can be set to arbitrary values, but it was found that the calibration yields good results for target firing rates of around one tenth of the input frequency. The firing rates are roughly brought close to the target firing rate with a bisection method searching for appropriate values of the parameters I_{fall} and afterwards fine tuned using a bisection method to find I_{out} for each synapse line driver.

It is important that the measured firing rates are averaged over multiple runs to prevent the bisection methods from jumping into wrong intervals because of statistical fluctuations in the firing rates. The calibration of the left side follows the same principle only the parameters for the right side are set to the values obtained by the bisection methods instead of the default values. After the calibration of the left side, the calibration for the right side is repeated and the values for I_{fall} and I_{out} of the left drivers are the ones obtained by the previous calibration step.

2.3.2 Calibration of Synaptic Weights

The algorithm described above assumes, that the impact of two inputs with a certain weight is approximately the same as the impact of one input with doubled weight. This is not the case for all neurons and therefore an additional calibration step is necessary. In this step the neurons are presented with two signals with an ITD of zero and their firing rates (averaged over multiple runs) are compared to the target firing rate used in the first step. The algorithm decides for each neuron whether its firing rate is too low or too high compared to the target firing rate and respectively increases or decreases the synaptic weights for the left and right side by one. This process is repeated 10 times.

2.3.3 Calibration Results

Figure 9 shows an exemplary calibration result for 3 detectors with input frequencies of $70\,\text{Hz}$, $100\,\text{Hz}$ and $130\,\text{Hz}$. After calibration the network was presented with signals of different ITDs and the mean firing rate of all neurons was plotted over the ITD. Figure 9 shows that the calibration algorithm is able to select synaptic strengths for which all neurons show an approximately equal firing behaviour with a maximum firing rate for an ITD of zero. These results were obtained by applying a shift factor of $s = 0.7$ to the input signals. A detailed compilation of the used network and neuron parameters can be found in table 3 in appendix A.

Figure 10 shows the results of the same experiment on the same network using the calibration parameters obtained by the calibration above for a shift factor of $s = 0.9$. It is important to note that for both shift factors no spike loss occurs. For the second shift factor the dependency of the firing rates on Δt changed strongly although in theory the shifting of the inputs should not have any influence on the firing behaviour of the network. The firing behaviour demanded in the Jeffress model is lost completely. Also the maximum firing rates changed from $10\,\text{Hz}$ to up to $80\,\text{Hz}$.

2.4 Interaction between Input Signals

The comparison of the figures 9 and 10 suggests that there is a dependency of the firing rates on the shift factor s. This dependency should theoretically not exist and indicates an interaction between the input signals for different neurons. To investigate it in more detail the dependency of the firing rate on s was measured with the same setup as in section 2.2.

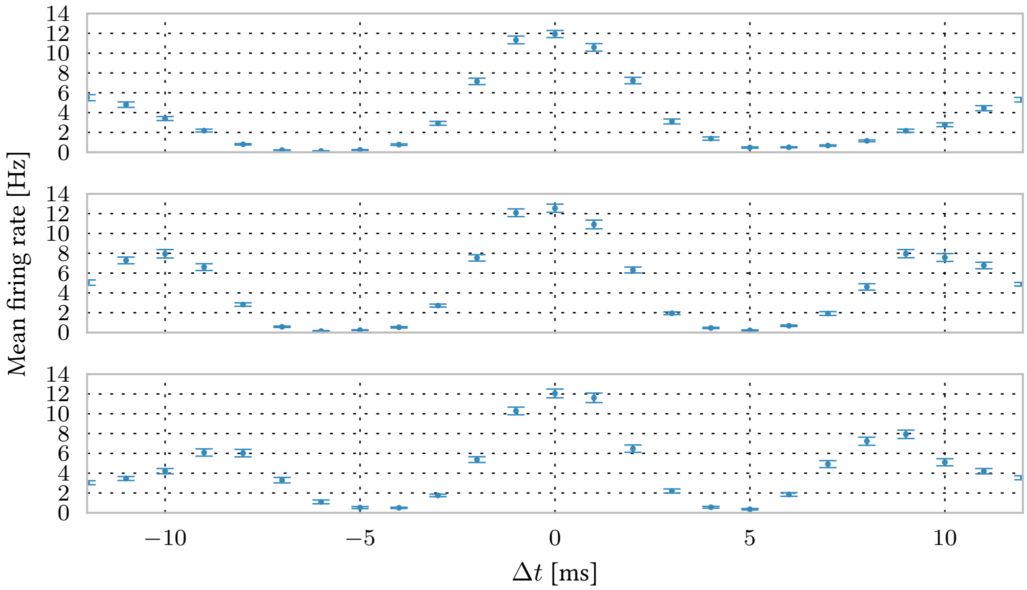

Figure 9: Calibration result for 3 detectors consisting of 41 neurons each calibrated to a maximum firing rate of 10 Hz. The plot shows the mean firing rates averaged over all neurons in a detector, plotted over the ITD between their inputs. The errors correspond to the standard deviation of the mean value. The input frequencies of the detectors are 70 Hz (top), 100 Hz (middle) and 130 Hz (bottom). The applied shift factor s was $s = 0.7$. The network consists of 41 neurons per detector instead of the maximum number of 42 neurons, because for an odd number of neurons it is possible to assign characteristic delays symmetrically around the value zero as well as to assign the delay zero to a neuron.

This was necessary, because for some shift factors spike loss occurs and only firing rates that are measured for no spike loss are comparable. Figure 11 shows the averaged firing rate of a network also used in figure 6. A large difference in the firing rates for different shift factors can be found, although no spikes are lost for either of them. The averaging over all neurons yields very large standard deviations of the mean value. This shows that the individual neurons respond differently on the changing time shift.

Figure 12 shows the investigation of two shift factors $s_1 = 0.5$ and $s_2 = 1.0$, both causing no spike loss, in detail. The firing rates for the individual neurons are significantly larger for the smaller shift. An exemplary membrane trace shows that this is caused by much higher PSPs for s_1.

The effect described above corrupts the functional principle of the Jeffress model, since the model assumes that the firing rate of a neuron is only dependent on the input frequency and the ITD between the left and right signal. The firing behaviour demanded by the model can be achieved by calibration as shown in figure 9. But the calibration prepares the network for one specific shift factor i.e. one specific timing of the input spike trains. If the input rate is preserved but the timing of the input is changed by a changing shift factor, the firing behaviour of the neurons changes strongly as shown in figure 10. Unfortunately, because of the added characteristic delays of the neurons and the ITD of

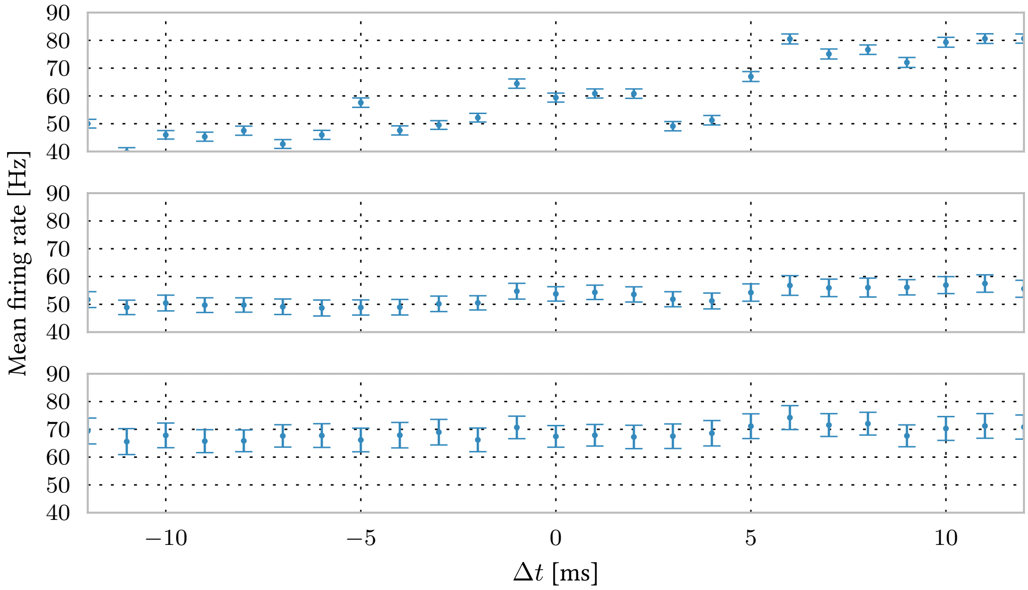

Figure 10: The same network and calibration as used in figure 9 with a changed shift factor of $s = 0.9$. Again the mean firing rates, averaged over all neurons in a detector, was plotted over the ITD between their inputs. The changing of the shift factor causes a strong changes in the firing behaviour of the neurons.

the input signals itself, this change in timing also happens, when the network is used to detect ITDs. Therefore the network in its current form is unable to detect an ITD. To bypass the unintended dependency on time shifts the mechanisms causing this effect have to be investigated and understood.

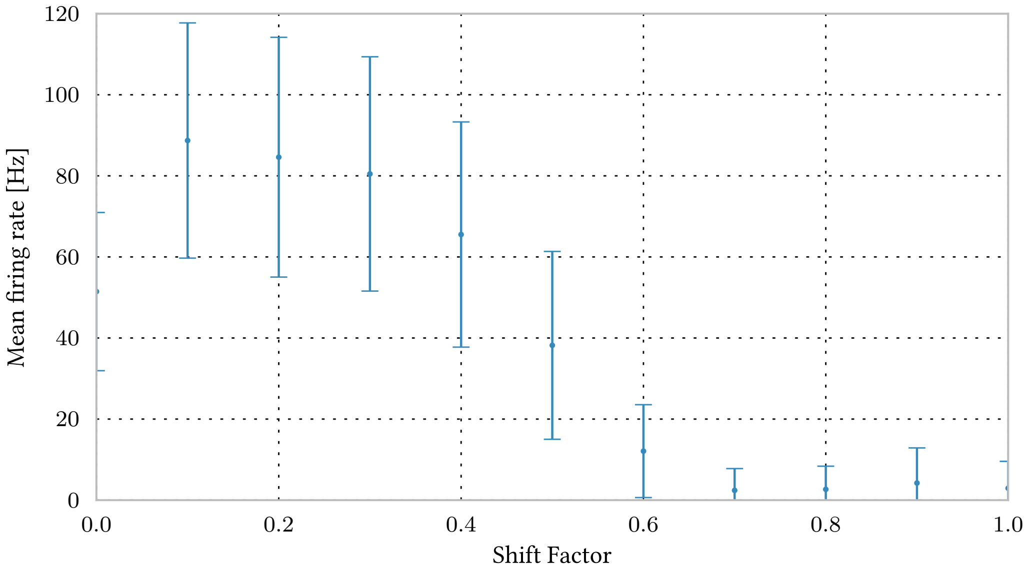

Figure 11: Mean firing rate of all neurons and standard deviation of mean value over the shift factor for 70 neurons with an input frequency of 100 Hz. For a shift factor of $s = 0.3$ as well as $s = 1.0$ there is no spike loss (see figure 6), but nevertheless the mean firing rate is around 80 Hz for $s = 0.3$ and close to 0 Hz for $s = 1.0$.

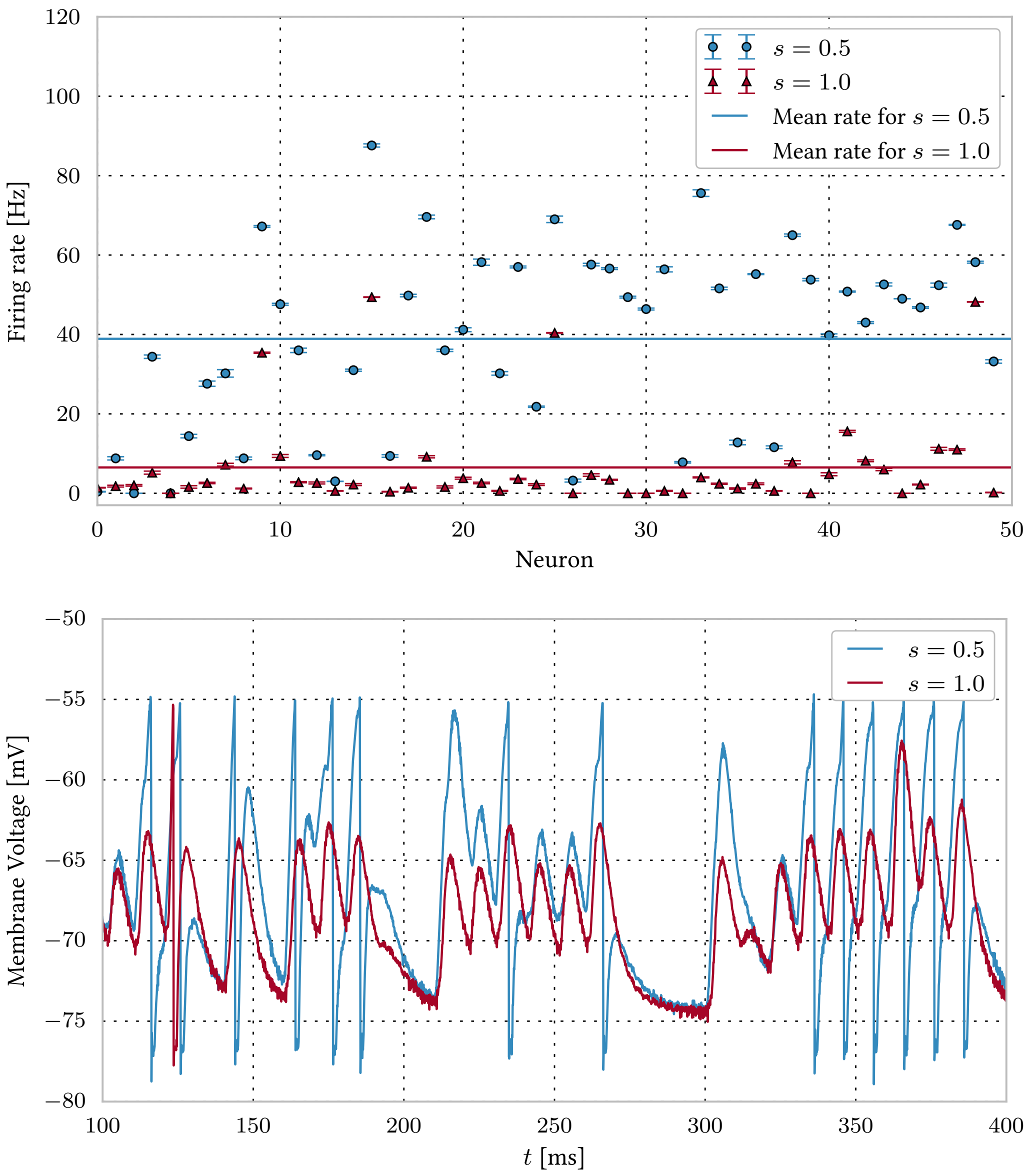

Figure 12: Influence of shift factor s on behaviour of neurons. Top: Firing rates of each neuron for two different shift factors, both having no spike loss. For $s = 0.5$, the rates are significantly higher. Bottom: Exemplary membrane traces of neuron number 42. The PSPs for the shift factor $s = 0.5$ are significantly higher which results in a higher firing rate for the neuron.

3 Investigation of Interaction between Input Signals

An input signal sent into a network by one synapse line driver seems to influence a second signal that is sent by another synapse line driver. If the second driver is connected to a neuron the first driver is not connected to, the impact of the second signal on that neuron is influenced by the first signal. By that the firing rate of a neuron is changed by an input signal that is not connected to the neuron. This effect seems to be dependent on the time relations between the two interacting signals. As the dependency of the described effect on the number of active synapse line drivers, the frequency, or the regularity of the signals might yield information about the underlying cause, it was investigated in detail and as isolatedly as possible.

3.1 Measurements

The network in figure 13 gives a measurement setup to investigate the influence of the number of active synapse line drivers, the spatial distance of the signals pathways, the timing relations between input signals, the parameters of the synapse line drivers and the input frequency on the described systematically.

One randomly chosen synapse line driver (*measurement driver*) is connected to all but one neuron and fed with a regular input of a fixed frequency. Symmetrically around that driver other drivers (*disturbance drivers*) are connected to the last remaining neuron and fed with regular input of the same frequency. The input spike trains of the disturbance drivers are shifted by a time interval Δt compared to the spike train for the measurement driver. A spatial distance between the measurement driver and the disturbance drivers is created by leaving a number of drivers without an input signal. By averaging the firing rates across all neurons and across randomly chosen measurement drivers the impact of the described parameters is determined systematically.

A detailed compilation of the network and neuron parameters used in the following can be found in table 4 in the appendix A.

3.1.1 Time Difference

Figure 14 shows the averages of all neurons' firing rates for an exemplary measurement driver and different numbers of disturbance drivers plotted over the time shift Δt between the signal of the measurement driver and the signals of the disturbance drivers. Already one disturbance driver on each side of the measurement driver causes differences in the firing rates of more than $10\,\mathrm{Hz}$ between $\Delta t = 12\,\mathrm{ms}$ and $\Delta t = 20\,\mathrm{ms}$. If more disturbance drivers are used, the differences in the firing rates increase. The dependency of the firing rates on Δt shows a periodicity of the same cycle duration as the input signals.

3.1.2 Comparable Measure

To compare the strength of the firing rates' dependency on the disturbing signals for different measurement drivers a comparable measure is needed. A suitable measure is the difference between the maximum of the mean firing rate $r_{\max}$, caused by an interaction

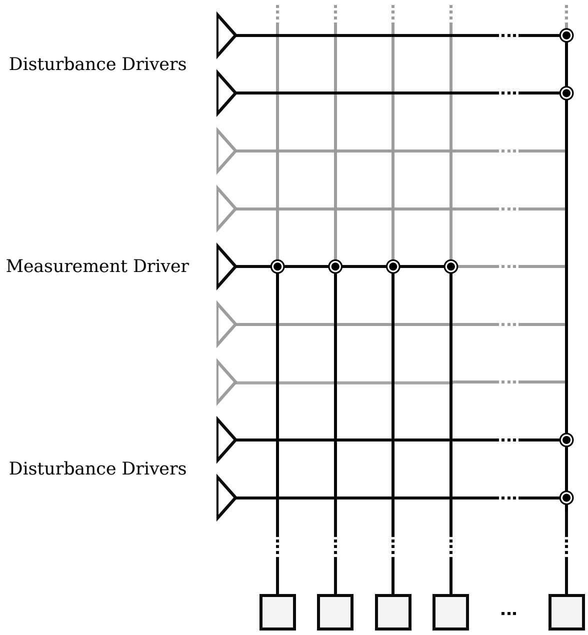

Figure 13: Network to investigate the interaction between the inputs. One measurement driver is connected to all but one neuron (squares) and fed with a regular spike train. Symmetrically around the measurement driver other drivers are connected to the last neuron. The spike trains sent to these drivers are shifted compared to the one of the measurement driver. The gray drivers are used to create a spatial distance and receive no input.

with the signals of the disturbance drivers, and the mean firing rate of the neurons if there is no disturbance by other drivers in the network r_{ref}:

$$r_{\text{diff}} = r_{\text{max}} - r_{\text{ref}}$$

In Figure 14 the maximum of the neurons' mean firing rates for one disturbance driver is $r_{\text{max}} = 28\,\text{Hz}$ and occurs for $\Delta t = 12\,\text{ms}$ and $\Delta t = 32\,\text{ms}$. For three disturbance drivers it is $r_{\text{max}} = 38\,\text{Hz}$ and occurs for $\Delta t = 10\,\text{ms}$ and $\Delta t = 32\,\text{ms}$. The reference rate for both is $r_{\text{ref}} = 18\,\text{Hz}$. This yields $r_{\text{diff}} = 10\,\text{Hz}$ for one disturbance driver and $r_{\text{diff}} = 18\,\text{Hz}$ for three disturbance drivers. These rate differences show that the influence of 3 disturbance drivers on the neuron is larger than the influence of one disturbance driver, which is also clearly visible in figure 14. But by using the parameter r_{diff} instead of plots like figure 14, it is possible to average over the results obtained by many different measurement drivers.

3.1.3 Active Synapse Line Drivers

Figure 15 shows the impact of spatial distance between the signals of measurement driver and disturbance drivers. The strength of the signals interaction was measured by measuring the parameter r_{diff} for 30 randomly chosen measurement driver and averaging the obtained values. The spatial distance was created by leaving synapse drivers between the measurement driver and the disturbance drivers without input as shown in figure 13. The distance is therefore measured in the number of drivers between measurement and dis-

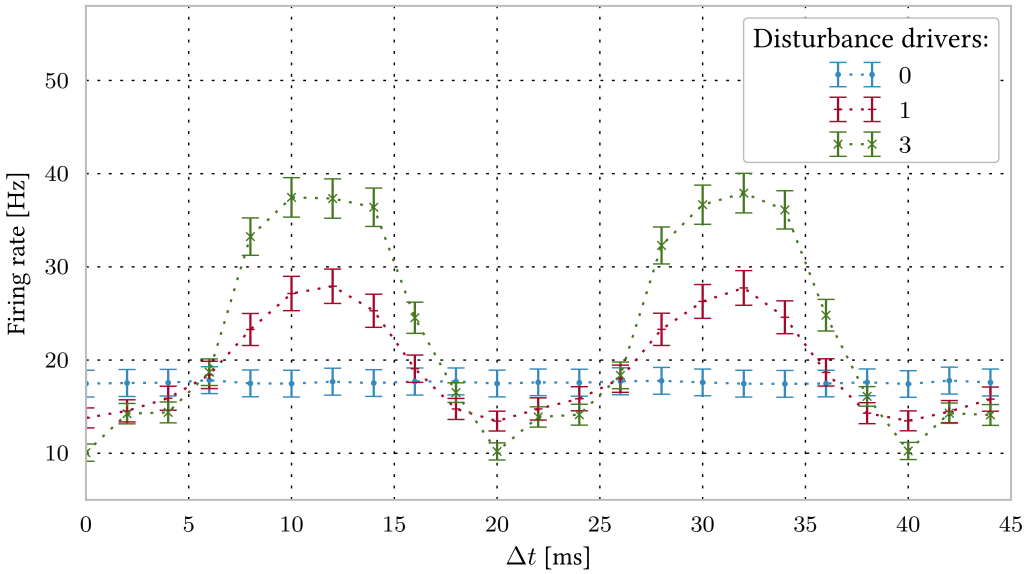

Figure 14: Dependency of the firing rates, averaged over all neurons, on the time difference Δt between the signals of measurement driver and disturbance drivers for a randomly chosen measurement driver (input frequency 50 Hz). As a reference the firing rates for a network without disturbance drivers are drawn.

turbance drivers. The results in figure 15 suggest, that the strength of interaction between the signals is not influenced by their spatial distance.

In Figure 16 the dependency of r_{diff} on the number of disturbance drivers, averaged over 30 randomly chosen measurement drivers, is shown. For an increasing number of disturbance drivers r_{diff} rises. This rise slows for larger numbers of disturbance drivers and saturates after approximately 30 drivers.

3.1.4 Hardware Current I_{rise}

The measurement of the dependency of r_{diff} on the number of disturbance drivers was repeated for a changed value of the hardware current I_{rise}. As described in section 1.3.2 I_{rise} controls the rising slew rate in the voltage ramp between synapse line driver and synapse. Its default value, which was used in the previous measurements, is $I_{\text{rise}} = 1.0\,\mu\text{A}$. In this measurement it was set to $I_{\text{rise}} = 0.1\,\mu\text{A}$. To achieve reasonably high firing rates, the neurons' resting potential was changed, too. Figure 17 shows, that for the lower I_{rise} the difference between reference rate and maximum rate also rises with the number of disturbance drivers but saturates at approximately 30 disturbance drivers. Compared to the measurement with $I_{\text{rise}} = 1.0\,\mu\text{A}$ the maximum values for r_{diff} are smaller for the smaller I_{rise}.

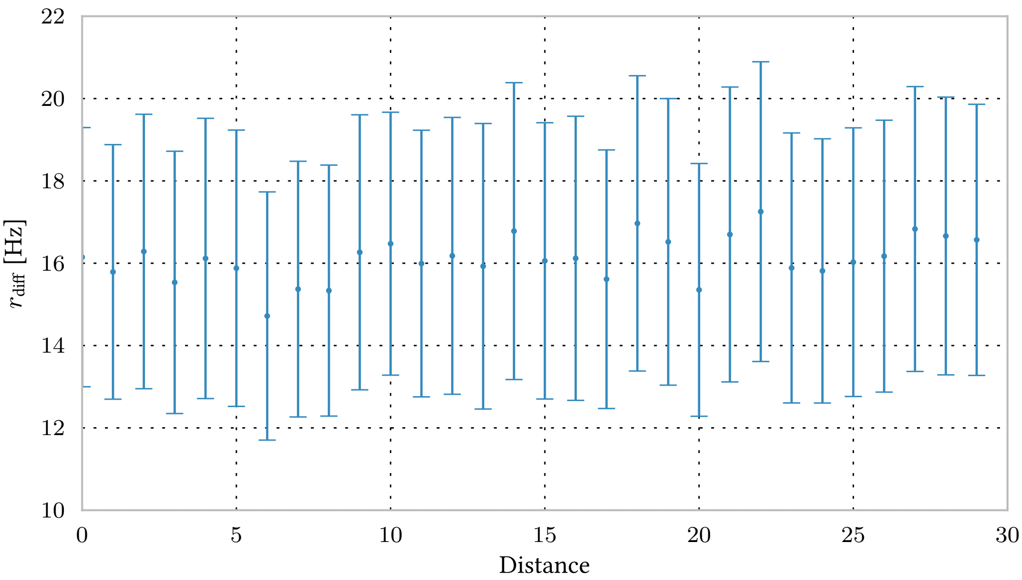

Figure 15: Dependency of r_{diff} on the distance between measurement and disturbance drivers for a number of 3 disturbance drivers and input signals with a frequency of 50 Hz. Each data point is the average over 30 randomly chosen measurement drivers.

3.1.5 Membrane Potential

The membrane potential of a neuron not receiving input from any synapse line driver was measured when only the measurement driver received an input signal with a frequency of 30 Hz. Figure 18 shows the obtained membrane trace. Additionally, the membrane potential of a neuron connected to the measurement driver is displayed. Ideally, the membrane potential of the first neuron should be approximately constant, but the membrane trace shows two prominent features: first, there are short, equidistant high frequency pulses occurring with the same cycle duration as the input signal. They are caused by crosstalk from the digital input to the read out electronics. For every incoming spike a short crosstalk signal is visible on the read out membrane voltage, although it is not actually occurring on the membrane (Schemmel et al., 2014). Since these pulses are only artefacts of the read out mechanism, they are not relevant for the investigated effect. Second, the membrane voltage decreases after every incoming spike. A comparison with the membrane potential of the second neuron shows, that the voltage drop last approximately as long as the rising flank of the PSP. In the appendix B the same measurement is repeated for an input frequency of 100 Hz (see figure 25). It shows that the periodic impact differs with the input frequency, since the membrane voltage in figure 25 shows periodic increases rather than decreases.

3.1.6 Additional Measurements

The experiments described above were repeated for the weights, connecting the disturbance drivers to the neuron whose firing rate is not measured, set to the value zero. The

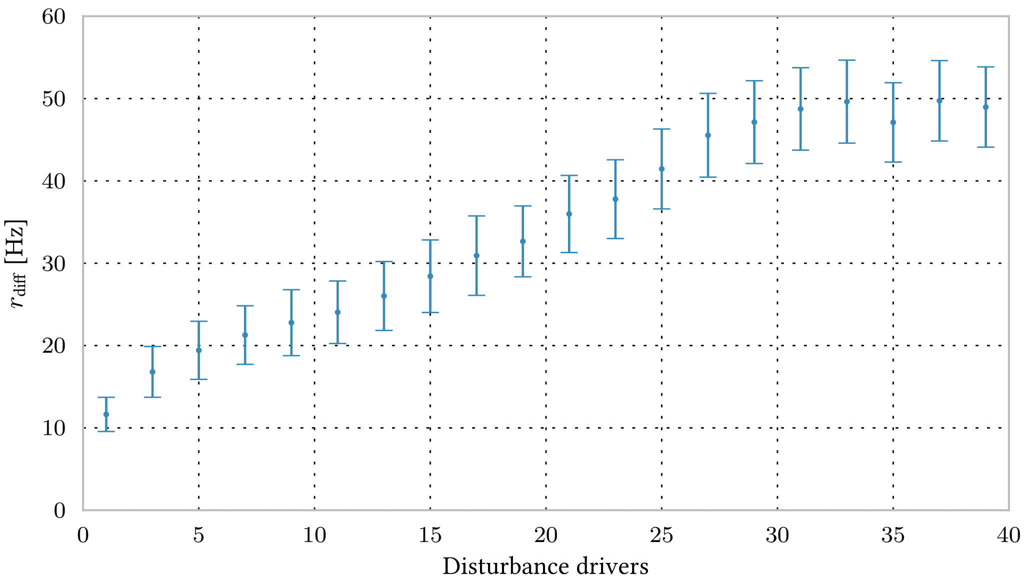

Figure 16: Dependency r_{diff} on the number of disturbance drivers for an input frequency of $50\,\text{Hz}$, a distance of 0 and $I_{\text{rise}} = 1.0\,\mu\text{A}$. Each data point is the average over 30 randomly chosen measurement drivers. The reference rate was approximately $r_{\text{ref}} = 8\,\text{Hz}$ for all data points.

results were the same as shown in the previous sections. The synaptic weight of the disturbance signals seems to have no influence on the interaction of signals.

More measurements concerning the dependency on the frequency and the networks behaviour when the regular spike trains are replaced by Poisson input as well as a comparison of the behaviour of all synapse line drivers can be found in the appendix B.

3.2 Conclusion

The most probable causes of the interaction between input signals are crosstalk in the synapse array or a temporarily overloaded power supply.

The observation that the synaptic weights of the disturbance signals seemingly have no influence on the interaction of the signals, suggests that the interaction is not caused by crosstalk in the vertical lines of the synapse array. If the signals interfered after the synapses, the strength of the interaction would scale with the synaptic weight, since the synapses scale the strength of the signals with the synaptic weight. Crosstalk in the horizontal lines of the synapse array can not be excluded by that fact, because the voltage ramps propagating between synapse line driver and synapse are independent of the synaptic weight. But crosstalk is rather short ranged and should happen only for signal pathways that are very close to each other. Figure 15 shows that the effect does not weaken with increasing distance as it would be expected for crosstalk.

An overloaded power supply could explain the observed effects, because a dropping supply voltage might change other voltages on the chip. These in turn might lead to changes in the effect of synaptic events on the membrane as well as to changes of the

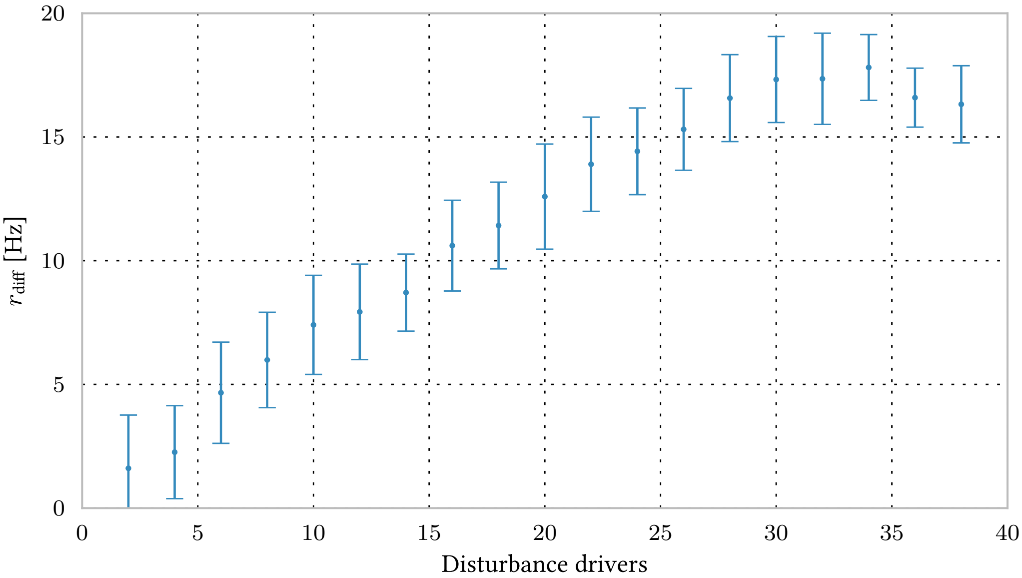

Figure 17: Dependency of r_{diff} on the number of disturbance drivers for an input frequency of 50 Hz, a distance of 0 and $I_{\mathrm{rise}} = 0.1\,\mu\mathrm{A}$. Each data point is the average over 30 randomly chosen measurement drivers. The reference rate was approximately $r_{\mathrm{ref}} = 12\,\mathrm{Hz}$ for all data points.

neurons' potentials. On the one hand, the membrane voltage of a neuron not connected to any input drops during the rising flank of a PSP of another neuron (figure 18). This suggests that the overload in the power supply may be caused by the high currents needed to generate the short rising flank of synaptic conductance (Schemmel et al., 2014). This hypothesis is supported by the fact that the maximum value of r_{diff} in figure 16 is larger than in figure 17 where I_{rise} was lowered. But these results have to be treated with care, since the reference rates for both experiments were not equal. If the input frequency rises or more synapse line drivers are used, the power supply might not be able to recover to the original value, which could lead to a change of other voltages and the described consequences. On the other hand, should a lower I_{rise} be less demanding for the power supply and therefore should the saturation of r_{diff} occur for a larger number of disturbance drivers as the saturation of r_{diff} for the larger I_{rise}. However, a comparison between figure 16 and 17 shows that r_{diff} saturates at the same number of disturbance drivers for both values of I_{rise}.

Summarized, it is not possible to state with certainty whether the power supply of the synapse line drivers is overloaded by the high currents during the generation of a synaptic event. In addition, it is not clear whether the power supply on the chip or on the board the chip is mounted on is responsible for the power shortage (Schemmel et al., 2014). To definitively locate the cause of the interaction between the input signals, further investigations are necessary.

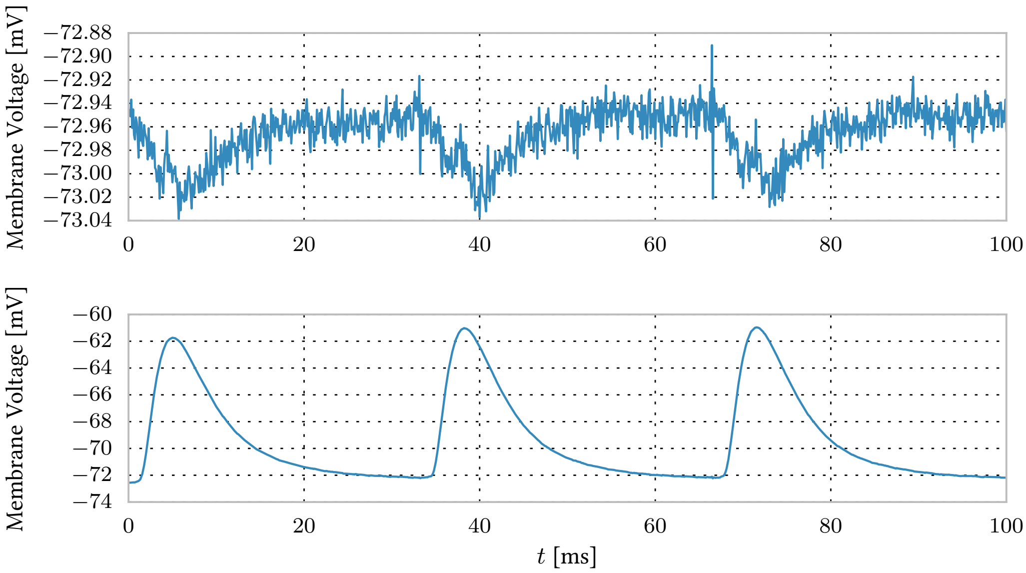

Figure 18: Top: Membrane voltage of a neuron receiving no input. The measurement driver was connected to other neurons and received an input signal of the frequency 30 Hz. Although the neuron should not see any input, there are short high frequency pulses, e.g. at $t \approx 66\,\mathrm{ms}$, as well as low frequency drops visible in the membrane voltage. Bottom: Membrane trace of a neuron connected to the measurement driver. Both traces are averaged over 100 emulation runs.

4 Single Detector Network

4.1 Network Architecture

The previous section showed that the unintended interactions between input signals for different neurons are caused by hardware properties that can not be changed or avoided in the scope of this thesis. It is therefore necessary, to change the network's architecture in a way that increases its robustness against these interactions that impair the network's performance.

The dependency of the strength of these unintended interactions, that is the increase of the differences in firing rates, on the input frequency shown in figure 24, suggests to reduce the input drastically. Then, the signals fed into the synapse array are distributed over a larger time interval, which should be less demanding for the power supply. The reduction of the input frequencies introduces the problem that the neurons, when receiving their input via only two synapse line drivers, are driven over their spike threshold far too seldom. Connecting two groups instead of single synapse line drivers representing the left and right input to each neuron compensates for the weaker input. This also has the positive side effect that each signal is sent to the neuron over multiple synapse line drivers. It is therefore possible to increase the probability p of thinning of the spike trains. By decreasing the input frequency and simultaneously increasing the thinning of the spike trains, the total amount of input for the network is reduced strongly. Additionally, the parameter I_{rise} that is responsible for the fast rise of the voltage ramp, which possibly causes the voltage drop, is reduced to $I_{\text{rise}} = 0.5\,\mu\text{A}$. Even lower values for I_{rise} were tested, but due to time constraints the optimal values for other parameters like the neurons' potential and the synaptic weights could not be found for them.

A detailed compilation of the network and neuron parameters used in the following can be found in table 5 in the appendix A.

4.2 Shuffling of Input Spike Trains

Because of bandwidth limitations the input spike trains can not be sent into the network at the same time for all neurons. Instead they are distributed over a certain time interval as described in section 2.2. To avoid that calibrations are valid for only specific timing of the input spike trains as described in section 2.3.3 the arrangement of the input spike trains is randomly drawn for every emulation run.

In detail, this means that no longer as shown in figure 5 the input spike train of the first synapse line driver starts first, the input of the second is shifted by one time step, the third by two time steps but the order is changed for every emulation run differently. It is essential, that the signals from the left and right ear to the same neuron are not shifted amongst each other, because otherwise the coding of the ITD in the phase difference between the signals is lost.

4.3 Calibration

Since the calibration algorithm developed in section 2.3 is only suitable for a network connecting each neuron to exactly two synapse line drivers, another algorithm is introduced here. The reason for the choice of I_{fall} and I_{out} as calibration parameters in section 2.3.1 was the too coarse resolution of the synaptic weight with only 16 possible values. Having now n synapse line drivers per input the number of possible values for the effective synaptic weight is increased to $n \cdot 16$. Since for $n > 3$ the resolution has shown to be large enough, the parameter tuned in the new calibration is the synaptic weight.

Like before it is necessary to calibrate all synaptic weights in parallel, because the calibration parameters are not completely independent of each other. In a first step all synaptic weights are set to an intermediate start value w_{start} and the firing rates of all neurons when receiving left and right input with an ITD of zero are measured. To minimize the effect of statistical fluctuations and the remaining interactions between the input signals, the firing rates are averaged over several emulation runs as well as over different random arrangements of the inputs spike trains. Then the firing rates are compared to a chosen target firing rate and for each neuron it is determined, whether the synaptic strength needs to be reduced or increased. If the firing rate is too high, the strongest synapse line drivers of left and right side are deduced and their synaptic weight is decreased by one. The identification of the strongest synapse line drivers is done by measuring the heights of single PSPs. For each neuron an input is created that sends single spike events to one synapse line driver after the other with large time intervals between the events. These events cause as many PSPs on the neurons membrane potential as there are synapse line drivers per neuron. By averaging over multiple repetitions of this protocol, the heights of the PSPs allow to sort the drivers by their strength. For a too small firing rate, the synaptic weights of weakest drivers are increased by one. After the correction of the weights for all neurons the new firing rates are measured and the process is repeated.

Unfortunately this calibration algorithm is strongly dependent on the start values of the synaptic weights w_{start}, the probability p of thinning as well as the target firing rate. Due to time constraints of this study, it was not possible to improve its stability. Nevertheless, calibration was satisfying, if the target firing rate was set to the mean value of the uncalibrated network's firing rates. This increases the dependency of the calibration result on the start value of the synaptic weights w_{start} since these weights are applied for the determination of the target firing rate. Figure 19 shows the strong influence of the start value of synaptic weight w_{start} on the calibration result. For $w_{\text{start}} = 3$ and $w_{\text{start}} = 4$ the target firing rate was chosen to the mean value of the uncalibrated network's firing rates. For the both parameter sets the neurons show the demanded firing behaviour, although the maximum rates show a difference of 11 Hz.

Since the calibration algorithm lacks the capability of automatically finding suitable parameters for p and w_{start}, these had to be found manually. For the detection of ITDs the firing behaviour achieved by $w_{\text{start}} = 3$ is most suitable, since the maximum is narrow and therefore the detection is likely to be precise.

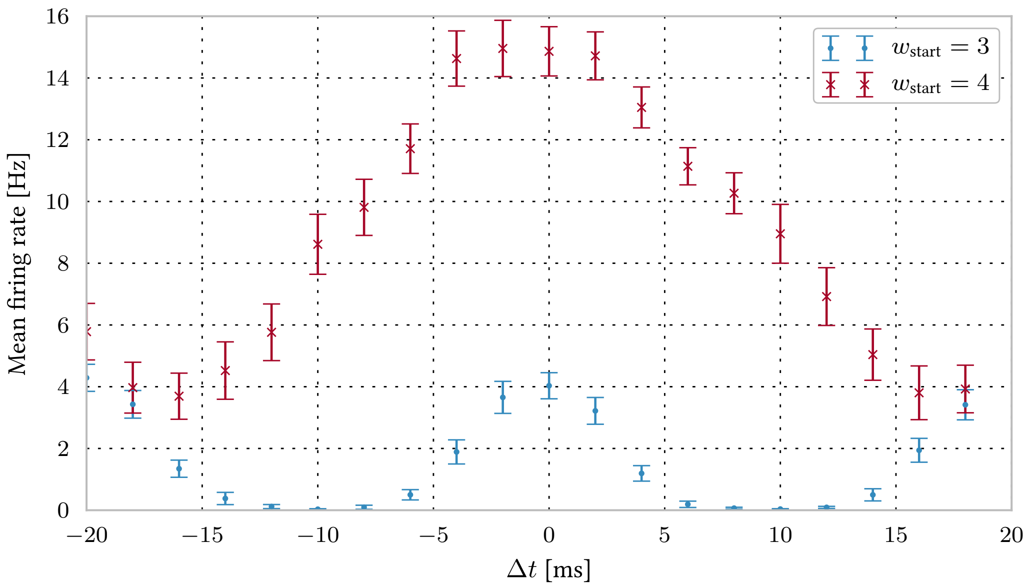

Figure 19: Calibration results for an ITD-detectors consisting of 20 neurons each connected to 4 left and 4 right input drivers for $w_{\text{start}} = 3$ and $w_{\text{start}} = 4$ both with a thinning of $p = 0.4$. The mean of the firing rates over all neurons is plotted over the ITD between their inputs. The errors correspond to the standard deviation of this mean value. The input frequency of the detector was 50 Hz.

4.4 ITD-Detection

In spite of the imperfect calibration and the possibly overloaded power supply the described network is used to detect ITDs. Figure 20 shows an example for an ITD-detection of input signals with a frequency of 70 Hz and an ITD of $\Delta t = -4$ ms. On the one hand the figure shows the averaged firing rates obtained by different arrangements of the shuffled input signals described in section 4.2. On the other hand the average over the different shuffle arrangements is depicted. This shows that the dependency on the timing of the inputs is still strong for single measurements, but the precision can be increased by averaging over different random arrangements of the input. Regarding the averaged response the network yields an approximately correct result. For an ITD of $\Delta t = -4$ ms the delay of $d = 4$ ms compensates the phase difference between left and right signal, and therefore the neuron corresponding to that delay fires with an higher rate. The other maxima are caused by the periodic input. For an input with the cycle duration $T \approx 14$ ms the ITD is also compensated by a delay of $d = -10$ ms and $d = 18$ ms. The maximum for $d = 18$ ms is shifted to slightly lower delays. A possible explanation for that might be that some neurons do not have their maximum firing rate for $\Delta t = 0$ ms but for slightly higher or lower delays. This might be caused by intrinsic delays in some of the synapse line drivers, e.g. due to variations in I_{rise}. In future studies a better calibration algorithm could filter out these neurons and drivers and replace them by others for which the maximum firing rate occurs for $\Delta t = 0$.

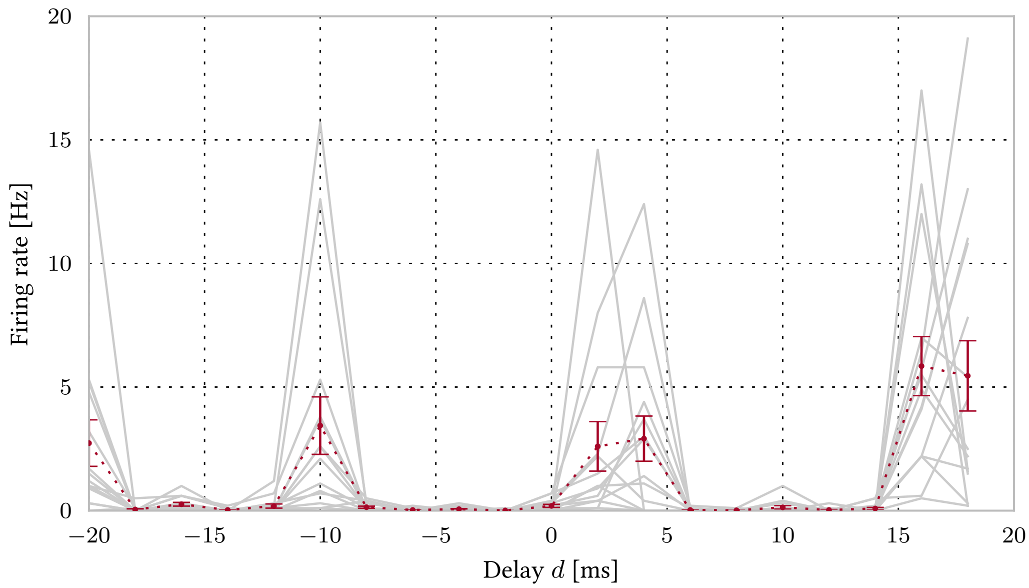

Figure 20: ITD-detection for input signals with an ITD of $\Delta t = -4\,\mathrm{ms}$ and a frequency of $70\,\mathrm{Hz}$. The firing rates of the neurons are plotted over the characteristic delays of the neurons. Each gray trace corresponds to one arrangement of the shuffled input signals described in section 4.2 (each averaged over 10 emulation runs). The average over the different arrangements (red) shows maxima for the expected delays.

The periodic response of the network shows, that multiple frequencies are necessary to resolve the ambiguity. But since each neuron in these network receives its inputs from more than two synapse line drivers, there are not enough drivers to run multiple detectors simultaneously on the chip. Consequently the experiment is split up in multiple runs for each detector.

4.5 Integration Layer

An integration layer as described in section 1.2 combines the results obtained by the different detectors. If the integration layer is used in a separate emulation run, the spike times of neurons in the detectors need to be recorded from the hardware and played back to the integration layer in another emulation run. Due to time constraints of this study this was not possible here.

As a proof of concept for the integration layer the recorded spike times of the neurons in the detectors can be substituted by a Poisson process of the same rate as the average rate across the different arrangements of input spike trains. To calibrate the integration layer the same algorithm as for the detectors was used. The only change was to use Poissonian input instead of regular spike trains and leaving the input unthinned ($p = 0$). Since the detectors consisted of 20 neurons, the integration layer uses the same number of neurons. Figure 22 shows that for Poissonian input the chip does neither show a strong dependency on the number of used synapse line drivers nor the timing of the inputs. Hence, all synapse line drivers can be used in an integration layer. Then each neuron is

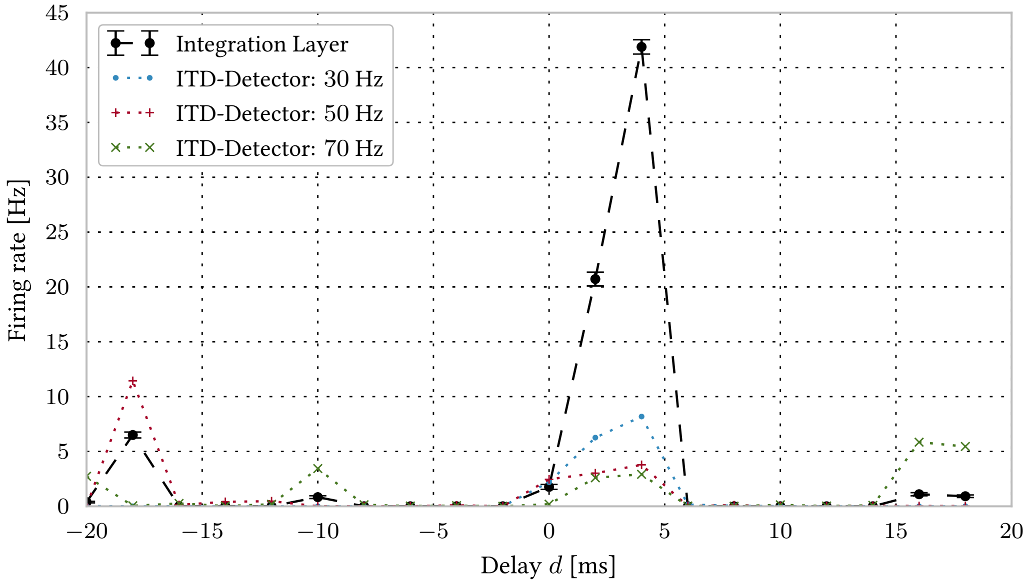

Figure 21: Exemplary result of an integration layer presented with Poisson input of the rates obtained in averaged measurements for the ITD-detectors (input frequencies of 30 Hz, 50 Hz and 70 Hz). For the sake of visibility the error bars were omitted. For 70 Hz see figure 20. The detectors were presented with signals with an ITD of $-4\,\text{ms}$. For the detectors the maximum rate at $d = 4\,\text{ms}$ as well as the maxima caused by the periodic input are visible. The integration layer damps the single maxima but strongly amplifies the common maximum of all three detectors. The result of the integration layer was averaged over 20 emulation runs.

connected to 12 synapse line drivers. Therefore, for an integration layer combining the results of 3 detectors, the firing rate of each detector neuron is sent in into the integration layer using 4 synapse line drivers. This makes the network more robust against parameter heterogeneity.

Figure 21 shows an exemplary result of such an integration layer combining and processing the results of 3 ITD-detectors with input frequencies of 30 Hz, 50 Hz and 70 Hz. Their input signals have an ITD of $\Delta t = -4\,\text{ms}$. The detectors show a common maximum for the delay $d = 4\,\text{ms}$ as well as additional maxima caused by the periodic input. The integration layer damps the single maxima but strongly amplifies the common maximum of all three detectors.

Note that the maximum rates of all detectors are be approximately equal. If one detector had much higher maximum firing rates than the others, this detector would overrule the results of the other detectors when processed by an integration layer.

5 Conclusion and Outlook

In this thesis the feasibility of performing binaural sound localisation on the neuromorphic microchip Spikey was investigated. In order to implement a network consisting of multiple ITD-detectors (Jeffress, 1948) on hardware, calibration algorithms compensating for leakage currents and inhomogeneities in the electrical components were developed. Futhermore, a method to overcome the limited input bandwidth of the chip by shifting the input signals was introduced. By applying these methods, all neurons could be calibrated to exhibit the firing behaviour required by the Jeffress model. However, in further experiments it was found that the required firing behaviour only occurs for the specific way of shifting the input signals used during calibration. Changing the temporal order or distance of theoretically independent input signals also changes the firing behaviour. This leads to the conclusion that there is a strong interaction between these signals. Since the Jeffress model assumes that the firing rate of a neuron only depends on the ITD between its inputs, this interaction between the signals impairs the detection of ITDs.

Regarding the strong impact of the described interaction on the network's firing rates, it seems astonishing that the effect was not relevant for other networks in previous studies (e.g. Pfeil et al., 2013a). However, further investigation showed that the interaction is not noticeable for sparse or random input. Even for high frequencies, large numbers of input sources and periodic signals, the effect is only noticed, if the results measured for different ways of shifting the inputs are compared systematically. To find the cause of the signal interaction, the dependency on parameters such as the number of used synapse line drivers, spatial distance of the signal pathways and hardware parameters was investigated systematically. The measurement results, already discussed in detail in section 3.2, exclude crosstalk effects in the synapse array and suggest an overloaded power supply as the most likely cause.

The knowledge obtained by the preceding investigations allowed targeted changes in the network to weaken the signals' interaction. Reducing the input frequency, lowering I_{rise}, connecting the neurons to more than two synapse line drivers and averaging over multiple different arrangements of the input spike trains enables the network to perform a successful ITD-detection. This serves as a proof of concept that a successful ITD-detection can be performed and may be improved if the overloading of the power supply is reduced. However, a systematical investigations of the network's performance are subject to further studies.

To consolidate the hypothesis of the overloaded power supply, further investigations are required. Measurements of the supply voltages not accessible through the software interface could reveal whether there is a voltage drop in the support electronics and whether it affects the voltage supply of the chip or the board the chip is mounted on. A possible voltage drop on the board might be avoided by improving or replacing the power supply. If the power drops within the chip, simulations of the hardware circuits can yield information necessary to improve the chip's design.

Furthermore, the calibration algorithm for the single detector network could be improved. Particularly, the dependency of the calibration result on the start parameter of the synaptic weights and the thinning of the spike trains needs to be reduced. The algorithm could also include a detection of synapse line drivers and neurons whose maximum firing rates occur at $\Delta t \neq 0$ because of intrinsic delays, most likely caused by parameter variations. These drivers and neurons can be blacklisted and replaced.

The necessity to average over multiple emulation runs does not affect the possible usage of the described networks e.g. in a robotic system, since the hardware is highly accelerated, which leaves enough time for the repetition of emulations. However, the costly computation of the characteristic delays for the neurons in software is not suitable for an application of the network in robotics. On the new version of the Spikey chip, the second neuron and synapse block is accessible. With the doubled number of neurons and synapse line drivers it could be possible to generate the delays using synfire chains and to run the detector network in parallel (Stoeckel, 2014; Pfeil, 2014). This would make the network less dependent on the controlling software and utilizable for practical applications.

A Emulation Parameters

Scope	Parameter	Value/git-hash
Hardware	Spikey-Version	4
	Spikey-Station	603
Software	PyNN-Version	0.6-hw
	symap2ic	26e2950a
	SpikeyHal	00755bc1
	vmodule	dbba1d6d

Table 2: General specification of used hardware and software.

Scope	Parameter	Value
Neurons	g_l	50 nS
	V_{rest}	-75 mV
	V_{th}	-45 mV
	V_{reset}	-75 mV
Network	Number of detectors	3
	Neurons per detector	41
	Input frequencies	70 Hz, 100 Hz, 130 Hz
	Synaptic weights	7
	Thinning p	0.3
	Simulation Time	2000 ms

Table 3: Network and neuron parameters for setup in section 2.

Scope	Parameter	Value
Neurons	g_l	$50\,\mathrm{nS}$
	V_{rest}	$-70\,\mathrm{mV}$
	V_{th}	$-50\,\mathrm{mV}$
	V_{reset}	$-70\,\mathrm{mV}$
Network	Disturbance drivers	3
	Synaptic weights	15
	Input frequency	$50\,\mathrm{Hz}$
	Distance	0
	I_{rise}	$1.0\,\mu\mathrm{A}$
	Simulation Time	$3000\,\mathrm{ms}$

Table 4: Default network and neuron parameters used for measurements in section 3. For the investigation of different dependencies one of these parameters at a time was changed.

Scope	Name	Value
Detector Neurons	Number of neurons	20
	g_l	$50\,\mathrm{nS}$
	V_{rest}	$-70\,\mathrm{mV}$
	V_{th}	$-45\,\mathrm{mV}$
	V_{reset}	$-70\,\mathrm{mV}$
Integration Layer Neurons	V_{th}	$-55\,\mathrm{mV}$
Detector 1	Input frequency	$30\,\mathrm{Hz}$
	Thinning p	0.3
	w_{start}	4
Detector 2	Input frequency	$50\,\mathrm{Hz}$
	Thinning p	0.4
	w_{start}	3
Detector 3	Input frequency	$70\,\mathrm{Hz}$
	Thinning p	0.3
	w_{start}	3
Integration Layer	Thinning p	0
	w_{start}	12
Network	Simulation Time	$3000\,\mathrm{ms}$

Table 5: Network and neuron parameters for setup in section 4.

B Additional Measurements

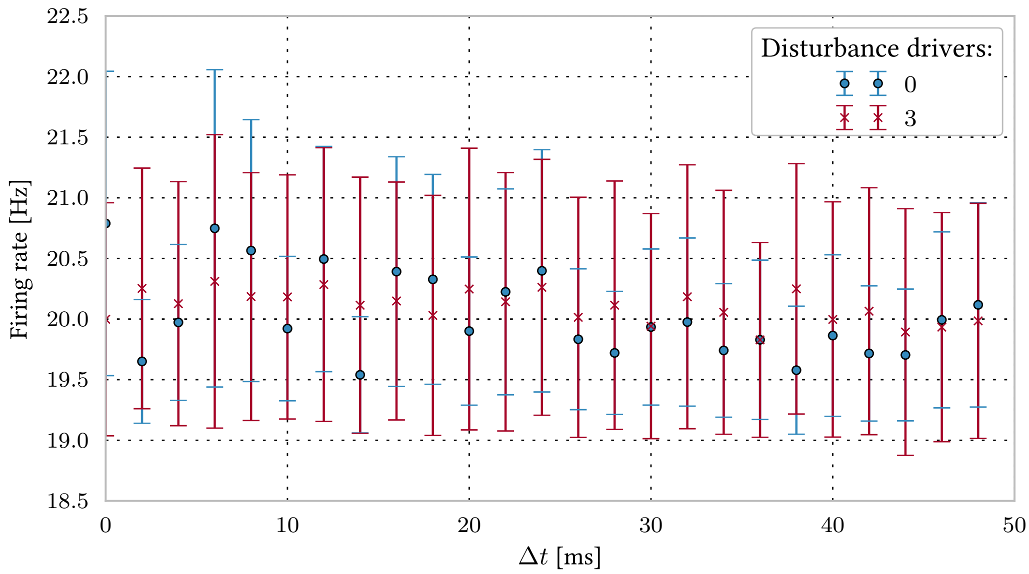

Figure 22: Dependency of the neurons' mean firing rates on the time shift Δt between the signals of measurement and disturbance drivers for a randomly chosen measurement driver when presented with spike trains drawn from a Poisson process. The dependency on the time shift does not seem to occur for randomly distributed spikes.

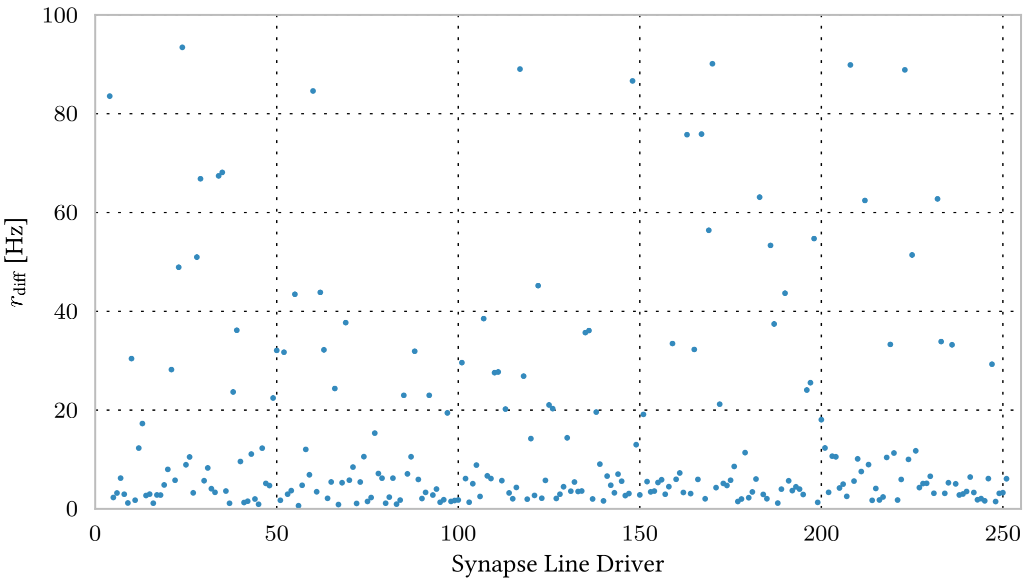

Figure 23: Measurement of r_{diff} for each synapse line driver. The input frequency was 50 Hz and the network consisted of 3 disturbance driver with no distance on each side of the measurement driver. The strength of the effect varies strongly for different drivers. There is no regularity visible. For the sake of visibility the error bars were omitted.

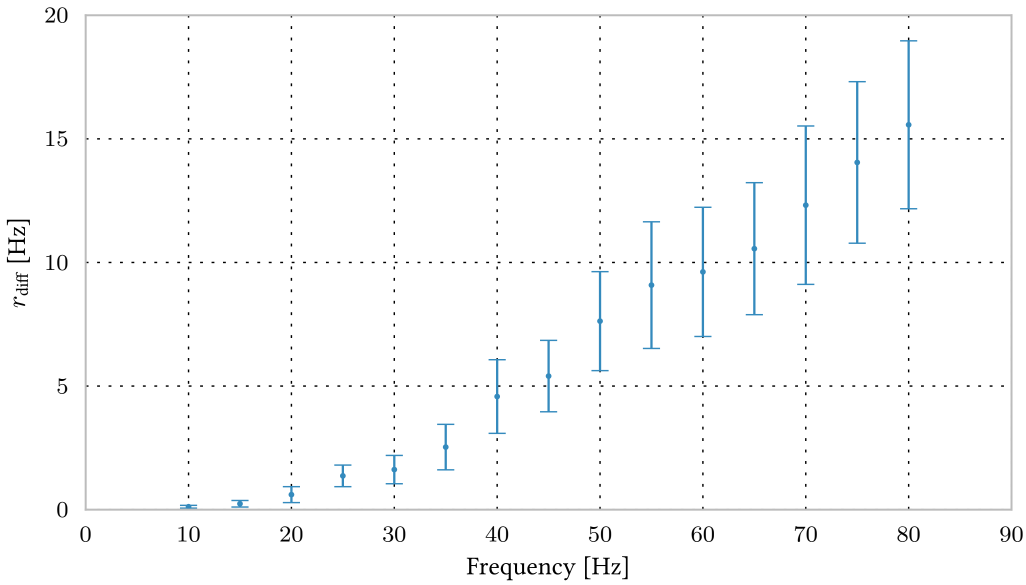

Figure 24: Dependency of r_{diff} on the input frequency. Each data point is the average over 20 randomly chosen measurement drivers. For the measurement 3 disturbance drivers on both sides with no distance between measurement and disturbance drivers were used. The strength of the effect grows with the input frequency. The variations between different measurement drivers increase as well.

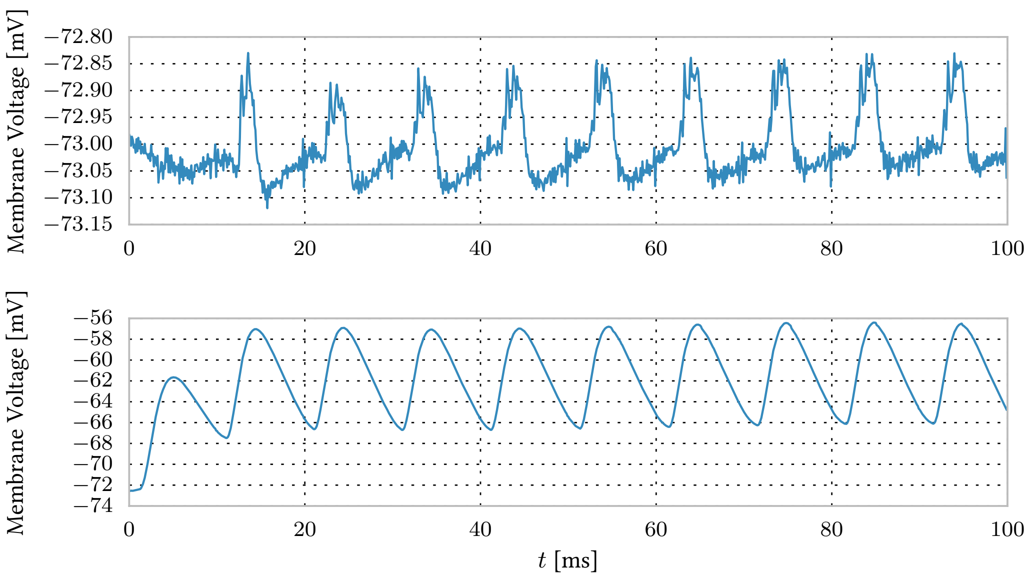

Figure 25: Top: Membrane voltage of a neuron receiving no input. The measurement driver was connected to other neurons and received an input signal of the frequency 100 Hz. Although the neuron should not see any input, there are short high frequency pulses as well as a periodic change in the voltage. Bottom: Membrane trace of a neuron connected to the measurement driver. Both traces were averaged over 100 runs.

References

BRAIN-Initiative. `http://www.nih.gov/science/brain/`, (accessed online: 2014-09-19), 2014.

BrainScaleS. `http://brainscales.kip.uni-heidelberg.de/public/index.html`, (accessed online: 2014-09-19), 2014.

D. Brüderle. *Neuroscientific Modeling with a Mixed-Signal VLSI Hardware System.* PhD thesis, Ruprecht-Karls-Universität Heidelberg, 2009.

C.E. Carr and M. Konishi. A circuit for detection of interaural time differences in the brain stem of the barn owl. *The Journal of Neuroscience*, 10(10):3227–3246, 1990.

A. P. Davison, D. Brüderle, J. Eppler, J. Kremkow, E. Muller, D. Pecevski, L. Perrinet, and P. Yger. PyNN: a common interface for neuronal network simulators. *Front. Neuroinform.*, 2(11), 2008.

P. Dayan and L. F. Abbott. *Theoretical Neuroscience: Computational and Mathematical Modeling of Neural Systems.* The MIT press, Cambride, Massachusetts, 2001. ISBN 0-262-04199-5.

FACETS. Fast Analog Computing with Emergent Transient States. `http://www.facets-project.org`, (accessed online: 2014-09-19), 2010.

W. Gerstner and W. M. Kistler. *Spiking neuron models: Single neurons, populations, plasticity.* Cambridge university press, 2002.

W. Gerstner, R. Kempter, J. L. van Hemmen, and H. Wagner. A neuronal learning rule for sub-millisecond temporal coding. *Nature*, 383:76–78, 1996.

M. Gewaltig and M. Diesmann. NEST (NEural Simulation Tool). *Scholarpedia*, 2(4):1430, 2007.

A. Grübl. personal communication, 2014.

Human Brain Project. `http://www.humanbrainproject.eu`, (accessed online: 2014-09-19), 2014.

L. A. Jeffress. A place theory of sound localization. *Journal of comparative and physiological psychology*, 41(1):35, 1948.

L. Kriener. Binaural sound localization in spiking neural networks. Internship Report, University of Heidelberg, 2014.

T. Pfeil. personal communication, 2014.

T. Pfeil, A. Grübl, S. Jeltsch, E. Müller, P. Müller, M. A. Petrovici, M. Schmuker, D. Brüderle, J. Schemmel, and K. Meier. Six networks on a universal neuromorphic computing substrate. *Frontiers in Neuroscience*, 7:11, 2013a.

T. Pfeil, A. Scherzer, J. Schemmel, and K. Meier. Neuromorphic learning towards nano second precision. In *Neural Networks (IJCNN), The 2013 International Joint Conference on*, pages 1–5. IEEE, 2013b.

J. Schemmel, D. Brüderle, K. Meier, and B. Ostendorf. Modeling synaptic plasticity within networks of highly accelerated I&F neurons. In *Proceedings of the 2007 IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 3367–3370. IEEE Press, 2007.

J. Schemmel, A. Grübl, and T. Pfeil. personal communication, 2014.

A. Scherzer. Phase-locking on neuromorphic hardware. Bachelor thesis (German), University of Heidelberg, HD-KIP 10-43, 2013.

D. Stoeckel. personal communication, 2014.

Statement of Originality (Erklärung):

I certify that this thesis, and the research to which it refers, are the product of my own work. Any ideas or quotations from the work of other people, published or otherwise, are fully acknowledged in accordance with the standard referencing practices of the discipline.

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

Heidelberg, September 26, 2014 ....

(signature)