Norwegian University
of Life Sciences

Master's Thesis 2019 30 ECTS
Faculty of Science and Technology

Existence and Detectability of Very Fast Oscillations in Spiking Network Models

Runar Helin
Data Science

Acknowledgments

I wish to thank my supervisor Hans Ekkehard Plesser for introducing me to the exciting field of computational neuroscience and for the guidance throughout the semester.

I also want to thank Simon Essink, Prof. Dr. Sonja Grün and Prof. Dr. Markus Diesmann for the idea for the topic of my thesis and all the interesting discussions.

Further, I want to thank Håkon Mørk and Stine Brekkes Vennemo for their assistance in running the simulations on computer clusters.

Finally, I want to thank Linda Josephine Claesson for her help and support throughout the semester as a fellow Data Science student writing a master thesis.

Abstract

Extremely high frequency oscillations are present in dynamics of different simulated neuronal networks, but are not yet observed in experimental recordings. This raises the question about the origin of these high frequency oscillations. Focusing on the microcircuit model of an area in the visual cortex, it is shown that the oscillations, visible as vertical stripes in raster plots from population activities, are related to peaks in the power spectra of the population-averaged firing activities. It is shown that the oscillations are not caused by the time discretization of the spike detection in the simulation of the network or the use of discretized delay values.

Given the difference in the number of neurons between experimental recordings and the populations in the network model, a subsampling of neurons from the network model with different sample sizes was performed to analyse the oscillations for subsamples of neurons with similar sizes as the experimental recordings. The results show that peaks in the power spectra can be observed for subsamples of few hundred neurons from a population, given that the whole population shows sufficient oscillatory activity.

Analytical results for the power spectra of the microcircuit model described by Bos et al. [2016] show that the choice of delay distribution for the inhibitory connections in the model has a large effect on the peaks. This network model originally uses a truncated Gaussian distribution for the synaptic delays. The use of an exponential delay distribution shows a spectrum with no peaks for frequencies above 100 Hz for any of the populations. Analytical power spectra from models using uniform and lognormal distribution for the delays have peaks in the power spectra around 80 Hz and 300 Hz, similar to the spectra from the model using a truncated Gaussian distribution. The amplitudes of the peaks are not the same for the models with different delay distributions. This is explained by the different contributions of each eigenmode obtained from the connectivity matrix of the network. It is also shown that varying the parameters of the delay distribution greatly shifts or scales both the low and high frequency peaks, making it possible to tune the parameters to manipulate the oscillations of the network dynamics.

The dynamics of area V1 from the multi-area model are analyzed for oscillations. The results show that the oscillatory dynamics are almost completely gone.

Sammendrag

Ekstremt høyfrekvente oscilleringer er tilstedeværende i dynamikken til ulike simulerte nevronale nettverk, men har enda ikke blitt observert i eksperimentelle målinger. Dette fører til spørsmålet om opphavet til disse høyfrekvente oscilleringene. Ved å fokusere på "microcircuit"-modellen for et område i synssenteret i hjernen vises det at oscilleringene, synlig som vertikale striper i raster plott fra populasjonsaktiviteten, kan kobles til topper i "power"-spekteret til den populasjons-gjennomsnittlige fyringsaktiviteten. Det vises at oscilleringene ikke oppstår på grunn av diskrete tidssteg i simuleringen av nettverket eller å bruke diskrete "delay"-verdier.

På grunn av forskjellen mellom antall nevroner i eksperimentelle målinger og størrelsen på populasjonene i nettverksmodellen, en "subsampling" av nevroner med ulike utvalgsstørrelser ble gjennomført for å analysere oscilleringene for utvalg med et antall nevroner i samme størrelsesorden som i de eksperimentelle målingene. Resultatene viser at toppene i "power"-spekteret kan bli observert for så lite som noen få hundre nevroner, gitt at populasjonen viser tilstrekkelig oscillatorisk aktivitet.

Analytiske resultater av "power"-spektrene til "microcircuit"-modellen, som forklart av Bos et al. [2016], viser at valget av "delay"-fordelingen til de inhibitoriske koblingene i modellen har en stor effekt på toppene. Denne nettverksmodellen bruker opprinnelig en trunkert Gaussfordeling for de synaptiske "delayene". Ved å bruke en eksponentiell fordeling til "delayene" observeres det ingen topp i de analytiske "power"-spektrene for frekvenser over 100 Hz. De analytiske "power"-spektrene for modeller med uniform- og lognormalfordelinger til "delayene" viser topper både for rundt 80 Hz og 300 Hz, tilsvarende resultatene fra å bruke modellen med den originale trunkerte Gaussfordelingen. Amplituden til på toppene i spektrene er forskjellig for modellene med de ulike "delay"-fordelingene som kan forklares ved de ulike bidragene til spektrene fra de ulike eigenmodene tilhørende koblingsmatrisen til nettverket. Det er også vist at variasjon i parametrene til delayfordelingene har en stor påvirkning på både posisjonen og amplituden til toppene i "power"-spektrene. Dette gjør det mulig å tune parameterne for å manipulere den oscillatoriske dynamikken i nettverket.

Dynamikken til området V1 av synssenteret i hjernen fra "multi-area"-modellen er analysert for oscilleringer. Resultatene viser at den oscillerende dynamikken er kraftig dempet.

Contents

Chapter 1

Introduction

Raster plots are a common way to visualize the dynamics of neuronal network models. Models such as the balanced random network [Brunel, 2000] and the microcircuit model by Potjans and Diesmann [2014] show regular periods with more synchronous firing activity of the neurons than others. These periods are visible as vertical stripes in the raster plots. The width of the stripes can be different for each model. In the microcircuit model used by Bos et al. [2016], stripes with a width of around 8 ms and stripes with a width of around 2 ms are observed (see Fig.1 in [Bos et al., 2016]). The stripes indicate oscillatory activity in the network dynamics. The stripes with a longer time width occur less frequently than the ones with smaller width. Therefore, the oscillations will be referred to as the low frequency and high frequency oscillations respectively.

The fast oscillations have not been observed in recored experimental data and raise the question whether theses oscillations occur due to some artifacts in the network models or if they are a property of network in the brain not yet observed. Analysis of oscillations in network models have been studied previously by Brunel and Hakim [1999], Bos et al. [2016], Brunel [2000], Buzsáki and Wang [2012] among others. Their results identifies connections that are responsible for the observed oscillations, and include suggestions for how these oscillations can be reduced by altering certain parameters like the strength of the connection between neurons and the delay of the signal between them.

Taking the microcircuit model [Potjans and Diesmann, 2014, Bos et al., 2016] as a basis, simulations from these network models with modified neuron models are analyzed to see if simplifications of the computations during the simulations causes the high frequency oscillations to appear.

In experimentally recorded data, there is a limit in the current technology of how many individual neurons it is possible to simultaneously record. Brochier et al. [2018] recorded around 150 single neurons simultaneously from a monkey trained to perform a well defined task. The stripes are not visible in the raster plot from this data, but the number of neurons is considerably lower than the number of neurons in the microcircuit model. Therefore, a subsampling of different sample sizes was performed on the simulated data to see how the detectability of the stripes changes with the sample size.

Analytical results of the network dynamics in the microcircuit model are presented in Bos et al. [2016]. Here, I use the same approach and analyses the analytical results using different distributions for the delays of the signals between neurons. The goal is to understand more of how the oscillations depends on the delays. It has been shown previously that the choice of the delays affects the oscillations in a simulated neuronal network [Brunel and Hakim, 1999]. They used a single layer network in their analysis, but the microcircuit model is multilayered. The goal here is to get a better understanding of how the analytical results of the oscillations change with alterations of the delay distribution of this network.

Finally, data from a more complex model, the multi-area model [Schmidt et al., 2018b] composed of several microcircuit models corresponding to different areas of the brain, is analyzed to see whether the oscillating dynamics are observed in this model.

Chapter 2

Theory

2.1 Brain basics

Neurons are the cells that make up the brain. They are responsible for the transmission of signals from the brain to other parts of the body [Thompson, 2000, ch. 2, p. 29–51]. A common estimate of the number of neurons in a human brain is 100 billion, but this number is debated [Lent et al., 2012]. Azevedo et al. [2009] shows that a human adult male brain consists of roughly 86 billion neurons. The neurons are connected in complex networks, each of which controls different parts of the body. The following description of the neuron and how it can be modeled is based on [Sterratt et al., 2011, ch. 2, p. 13–46].

A neuron is composed of a *soma* or cell body, *dendrites* and an *axon*. Neurons sends signals along its axon and receives signals from other neurons at the dendrites. A cell membrane separates the extracellular and intracellular space of the neuron and is impermeable to ions except at different ion channels. The potential difference between the extracellular and the intracellular areas are called the *membrane potential* and is typically around -65 mV in its resting state.

A neuron receives signals from other neurons at the dendrites via synapses. These synaptic inputs activates the ion channels, letting ions flow in or out of the extracellular space which affects the membrane potential. The increase and decrease of the membrane potential is called *depolarization* and *polarization* respectively. Neurons is often divided into two groups, *excitatory* and *inhibitory* neurons. Signals from excitatory neurons result in depolarization while signals from inhibitory neurons result in polarization. Given enough excitatory signals, the membrane potential of a neuron will cross a certain threshold value. When this happens, a feedback effect causes the

membrane potential to increase rapidly before it decays back to its resting membrane potential again. This signal is called an *action potential* and is important for the transmission of information between the neurons. The action potential travels down the axon of the sending neuron and transfers the signal to other neurons at the synapses.

This current of ions through the cell membrane is dependent on the membrane potential. The current can be approximated as the current driven through a resistor by an electromotive force described the equation

$$I_{\mathrm{i}} = \frac{1}{R_{\mathrm{m}}}(V - E_{\mathrm{m}}), \tag{2.1}$$

where I_{i}, is the ion current, R_{m} is the membrane resistance, V is the membrane potential and E_{m} is the electromotive force driving the current.

The impermeability of the cell membrane to ions causes a build-up of charge on either side of the membrane. Therefore, the membrane can be considered a capacitor. The capacitive current is defined by

$$I_{\mathrm{c}} = C_{\mathrm{m}}\frac{dV}{dt}, \tag{2.2}$$

where I_{c} is the capacitive current, V is the membrane potential and C_{m} is the membrane capacitance.

By describing the cell membrane by these equivalent electrical components, the dynamics of the membrane potential in response to an input current I_{e} can be modeled as an RC-circuit. Given a section of a cell membrane with surface area a, the dynamics of the membrane potential are described by the following differential equation.

$$C_{\mathrm{m}}\frac{dV}{dt} = \frac{E_{\mathrm{m}} - V}{R_{\mathrm{m}}} + \frac{I_{\mathrm{e}}}{a}, \tag{2.3}$$

The constants C_{m}, E_{m} and R_{m} are determined experimentally.

2.2 Neuron models

Models of neurons are used to increase the understanding of how neurons work. Results from a model can be compared to experimental data and in that way be used to test different hypotheses. A wide range of different neuron models exists, some of which are more detailed than others. More detailed models takes different ion channels, spatial distribution of dendrites and axons, and intracellular mechanics among other things into account. The choice of neuron model in a study depends on the goal. A classical

neuron model is the Hodgkin-Huxley model (HH model) which was the first model to accurately describe the action potential [Sterratt et al., 2011, ch. 3, p. 47–71]. In this model, the ion channels are described by more detailed equations. They included two different ion channels in their model, one for sodium and one for potassium ions. Based on experimental data obtained from a giant squid axon, they modeled the ion channels as gates controlled by gating variables whose dynamics were described by differential equations.

When studying the dynamics of networks of neurons, it is advantageous to use a simpler neuron model in order to make the simulation computationally feasible, and to be able to include enough neurons so that the modeled network behaves realistically compared to biological neuronal networks.

2.2.1 Leaky integrate-and-fire neuron (LIF)

The network analyzed in this thesis is composed of *leaky integrate-and-fire* (LIF) neurons. The description of this neuron model is based on the description by [Sterratt et al., 2011, ch. 8, p. 196–225].

The Leaky integrate-and-fire neuron model is a simplified neuron model where only the subthreshold dynamics of the membrane potential is modeled and not the action potential itself. The subthreshold dynamics obeys the following differential equation

$$\frac{dV}{dt} = -\frac{V}{\tau_{\mathrm{m}}} + \frac{I(t)}{C_{\mathrm{m}}}, \tag{2.4}$$

where V is the membrane potential, τ_{m} is the membrane time constant, C_{m} is the membrane capacitance and I is the input current. The membrane time constant τ_{m} is defined as $\tau_{\mathrm{m}} = R_{\mathrm{m}}C_{\mathrm{m}}$, where R_{m} is the membrane resistance.

To describe the input currents to a LIF neuron, two common models are used. conduction-based and current-based synaptic currents. The difference between the two types of synaptic currents is that for conduction-based synaptic currents, the time course of the *conductance* is described rather than the current itself. The conduction-based synapse is described by

$$I(t) = g_{\mathrm{syn}}(t)(V(t) - E_{\mathrm{syn}}), \tag{2.5}$$

where g_{syn} is the conductance and E_{syn} is the reversal potential defined as the potential for when the current changes direction. The reversal potential is a parameter of the

synapse that is modeled. A simple description of the time course of the conductance is that of a falling exponential

$$g_{\text{syn}}(t) = \begin{cases} \bar{g}_{\text{syn}} e^{-\frac{t-t_s}{\tau_{\text{syn}}}} & \text{for } t \geq t_s \\ 0 & \text{for } t < t_s \end{cases}. \tag{2.6}$$

where $\bar{g}_{\text{syn}}$ is the maximum conductance of the synapse and τ_{syn} again is the synaptic time constant.

A simple but often used model for the current-based synaptic input is a decaying exponential current

$$I(t) = \begin{cases} we^{-\frac{t-t_s}{\tau_{\text{syn}}}} & \text{for } t \geq t_s \\ 0 & \text{for } t < t_s \end{cases} \tag{2.7}$$

where w is the synaptic weight describing the strength of the synaptic connection and τ_{syn} is the synaptic time constant which determines how fast the current decays. The time t_s is the time when the signal arrives at the neuron. For $t < t_s$, the input current is zero and at t_s the current is at its maximum before it decays towards zero.

Synaptic currents cause the membrane potential of the neuron to change. When the membrane potential reaches a defined threshold value θ, the potential is reset to a resting value called V_{reset} for a time equal to the absolute refractory period τ_{ref}.

2.3 Microcircuit model

The microcircuit model is a neuronal network model of LIF neurons and was first proposed by Potjans and Diesmann [2014]. The model represents a network of neurons from four cortical layers named L2/3, L4, L5 and L6. Each of theses layers is populated by two types of model neurons, *excitatory* neurons and *inhibitory* neurons. Each population contains a different number of neurons and has a different number of connections to the other populations. Both the number of neurons in each population and the number of connections are determined from different experimental results. The experimental results come from both anatomically and electrophysiological studies. Details of how these results were combined for the microcircuit model are described by Potjans and Diesmann [2014] and will not be discussed in detail here.

Both neuron types in this model are LIF neurons with detailed parameters listed in Table 5 by Potjans and Diesmann [2014]. The number of neurons in each population are listed in Table 2.1. The two populations in each layer, the excitatory and inhibitory

neurons, will be referred to as LXE and LXI, where X is replaced by the layer number, i.e. L4E is used for the excitatory neurons from layer L4.

Table 2.1: Number of neurons in each population in the microcircuit model. The data is extracted from Table 5 in Potjans and Diesmann [2014].

Population	L2/3E	L2/3I	L4E	L4I	L5E	L5I	L6E	L6I	Total
Number of neurons	20,683	5,834	21,915	5,479	4,850	1,065	14,395	2,948	77,169

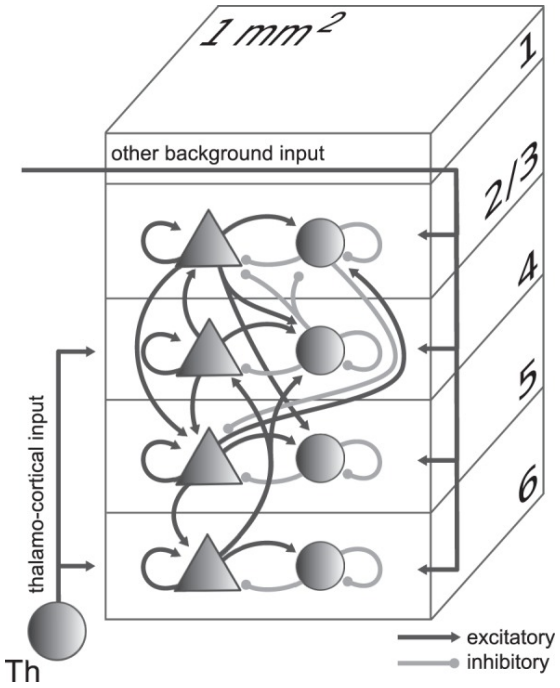


Figure 2.1: Representation of the microcircuit model. Each of the four layer consists of two populations illustrated by the triangles and circles. The network size represents that of a cortical network under a patch with surface area 1 mm^2. The figure is reprinted from Potjans and Diesmann [2014] under the Creative Commons Attribution License (http://creativecommons.org/licenses/by-nc/3.0/).

The size of the microcircuit model is chosen so that it represents a region of a cortical network under a patch with surface area 1 mm^2 as visualized in Fig 2.1. Neurons in the different populations are connected by synaptic connections both inside each layers and across the different layers. This results in a total of 64 possible projections. In this context projection is used to describe the connection from a source population to a target population.

From the experimental results Potjans and Diesmann [2014] defines an 8×8 matrix M called the connectivity map given in Table 2.2. The entries $M_{i,j}$ describes the

Table 2.2: Connection probabilities between the different populations in the microcircuit model. Data extracted from Table 5 in Potjans and Diesmann [2014].

To \ From	L2/3E	L2/3I	L4E	L4I	L5E	L5I	L6E	L6I
L2/3E	0.101	0.169	0.044	0.082	0.032	0.0	0.008	0.0
L2/3I	0.135	0.137	0.032	0.052	0.075	0.0	0.004	0.0
L4E	0.008	0.006	0.050	0.135	0.007	0.0003	0.045	0.0
L4I	0.069	0.003	0.079	0.160	0.003	0.0	0.106	0.0
L5E	0.100	0.062	0.051	0.006	0.083	0.373	0.020	0.0
L5I	0.055	0.027	0.026	0.002	0.060	0.316	0.009	0.0
L6E	0.016	0.007	0.021	0.017	0.057	0.020	0.040	0.225
L6I	0.036	0.001	0.003	0.001	0.028	0.008	0.066	0.144

probability that a specific neuron from population j is connected to a specific neuron in population i. The actual neurons that are connected are randomly drawn based on this connectivity map. In addition, the strengths of the synaptic connections are randomly drawn from a truncated Gaussian distribution with mean and standard deviation listed in Table 2.3. The parameter g is the factor used to increase the inhibitory synaptic strength and is larger than the excitatory synaptic weight to achieve a balanced random network, meaning that the total strength of the excitatory and inhibitory synapses to a neuron are equal even though the number of each type of connection is different. The inhibitory synaptic weight is negative since it polarizes the membrane potential.

Table 2.3: Parameters for the synaptic connections in the PD model. The data is extracted from Table 5 in Potjans and Diesmann [2014].

Name	Symbol	Value
Excitatory synaptic strength	$w \pm \delta w$	87.8 ± 8.8 pA
Inhibitory synaptic strength	$gw \pm \delta w$	$-4 \times 87.8 \pm 8.8$ pA
Excitatory delays	$d_e \pm \delta d_e$	1.5 ± 0.75 ms
Inhibitory delays	$d_i \pm \delta d_i$	0.8 ± 0.4 ms

In the microcircuit model, Potjans and Diesmann [2014] introduce heterogeneity in the network by assigning delays to the synaptic connections drawn from a probability distribution. The delays are drawn from a truncated Gaussian distribution with mean and standard deviation specified in Table 2.3. The delay values need to be positive to

make sense, negative delay values are therefore redrawn from the same distribution. Note that the delay parameters for the delay distribution are different for the inhibitory and the excitatory connections.

2.4 Network models

The analysis in this thesis is done using two versions of the microcircuit model. The first one is the microcircuit model with its original parameters as described in in Potjans and Diesmann [2014], and will be referred to as the PD model. This network is implemented in NEST and is included as an example script bundled with the software. The second one is almost identical except for some slight modifications that was introduced by Bos et al. [2016]. This model will be referred to as the BDH model.

Networks can operate in different states as described in Brunel [2000]. The two main states of interest in this thesis is the *synchronous irregular* (SI) state and the *asynchronous irregular* (AI) state. A network in the SI state shows oscillatory synchronous activity when looking at the activity of the whole network, but the activity of individual neurons are irregular. A network in the AI state on the other hand, shows a stable global activity and with irregular activity from individual neurons. It can be shown that the dynamics of the microcircuit model described by Potjans and Diesmann [2014] are close to that of an SI state. Because of this, Bos et al. [2016] stabilized the network by reducing the number of synaptic connections from L4I to L4E by 15 %, and increased the standard deviation of the delay distribution of all synaptic connections. In order to keep the firing rate the same, the external input to L4E was reduced to counteract the reduction of inhibitory connections to L4E. The differences between the BDH model and PD model are listed in Table 2.4.

Table 2.4: Parameter differences between the PD model and BDH model. The parameter $K_{4I,4E}$ is the number of connections from L4I to L4E determined by the number of neurons and the connection probability from Table 2.1 and Table 2.2.

	Connections L4I → L4E	δd_e	δd_i	**External input**
PD model	$K_{4I,4E}$	0.75 ms	0.4 ms	2100
BDH model	$0.85\,K_{4I,4E}$	1.5 ms	0.8 ms	1780

2.5 Statistical analysis

The action potentials are often referred to as *spikes*. In simulations of neuronal networks, the time each neuron fires a spike is recorded. The sequence of spike times for a neuron is called a *spike train*. This section briefly presents some of the standard statistical methods used to analyse spike trains.

2.5.1 Time histograms

Given a simulation of N number of neurons simulated for a time T. The time histogram is calculated by first dividing the total simulation time into time bins with equal bin size Δ and then count the number of spikes across all the neurons at each time bin. From this, one can get an estimate of the population averaged firing rate at each time bin with the following equation

$$r_i = \frac{n_i}{N\Delta}, \tag{2.8}$$

where r_i is the firing rate and n_i is the spike count at time bin i.

2.5.2 Auto-correlation and Cross-correlation

A spike train S_i can be defined by

$$S_i(t) = \sum_k^K \delta(t - t_k), \tag{2.9}$$

where K is the total number of spikes of S_i and $\delta(\cdot)$ is the *Dirac delta function*. The auto-correlation C_{ii} for the spike train is defined by

$$C_{ii}(\tau) = \langle S_i(t)\, S_i(t+\tau)\rangle_t, \tag{2.10}$$

where $\langle\,\rangle_t$ denotes the time average [Gerstner and Kistler, 2002]. The autocorrelation is a function of the time lag τ which describes how the spike times of the neuron correlates with the spike times the time τ later.

The cross-correlation on the other hand describes how well the spike times of one neuron correlate with the spike times of another neuron. Given two spike trains S_i and S_j, the cross-correlation is defined by

$$C_{ij}(\tau) = \langle S_i(t)\, S_j(t+\tau)\rangle_t. \tag{2.11}$$

2.6 Spectral analysis

The study of frequency components of signals is called spectral analysis, where Fourier transformations are an important tool. Fourier transforms are widely used in signal processing such as spectrum analysis and audio compression [Smith, 2008]. The purpose of Fourier transformations is to represent periodic signals as a sum of different frequency components. Oscillations observed in neuronal network activities can be analyzed with the use of power spectrua by looking at the power spectrum of the population averaged firing activity in the network. The frequencies of the oscillations will be visible as peaks in the power spectrum. Before describing how the power spectrum is obtained for a population averaged firing activity, the discrete Fourier transformation procedure will be presented.

2.6.1 Discrete Fourier Transformation

This section presents the method of the discrete Fourier transformation as described by Kreyszig [2011]. The discrete Fourier transform procedure is part of Fourier analysis that is applied to discrete functions. Similar to the standard Fourier transformation of continuous functions, the transformed signal data represents the frequency spectra of the original signal. The discrete Fourier transform is applied to sampled data points with equal spacing. For a signal f sampled by N equally spaced data points, the discrete Fourier transform is defined as

$$\hat{f}_n = \sum_{k=0}^{N-1} f_k e^{-ik2\pi n/N} \qquad \text{for } n = 0,\ldots,N-1, \tag{2.12}$$

where f_k is the k^{th} data point and $\hat{f}_n$ is the n^{th} transformed data point. The transformation can be written in vector notation as follows

$$\hat{\mathbf{f}} = \mathbf{F}_N \mathbf{f}, \tag{2.13}$$

where $\mathbf{f}$ is the vector of the sampled data, $\hat{\mathbf{f}}$ is the vector of transformed data and the element in row n and column k of $\mathbf{F}_N$ is $e^{-ik2\pi n/N}$. The matrix $\mathbf{F}_N$ is called the *Fourier matrix*.

The discrete Fourier transform of a signal can in principle be calculated by using equation 2.13, but for signals with many data points this procedure is very slow due to the large number of entries in the Fourier matrix which leads to a large number of calculations. An alternative implementation of the discrete Fourier transform is

therefore often used to overcome this challenge. The method is called the *fast Fourier transform* (FFT) and divides a problem of size N recursively into two smaller problems with size $N/2$ each until each problem is of size 2. The Fourier components are then found for each of the subproblems. The results of using the fast Fourier transform is exactly the same as using Eq. 2.13 and will therefore be used throughout this project. Algorithms for the fast Fourier transform are implemented in several Python libraries such as *NumPy* [Oliphant, 2006–].

2.6.2 Estimation of the power spectral density

Given a signal $\mathbf{f}$ sampled with N points and its FFT transformed signal $\hat{\mathbf{f}}$. The vector $\mathbf{f}$ belongs to the time domain and $\hat{\mathbf{f}}$ belongs to the frequency domain.

The vector of absolute values of the sampled data points squared is an estimate of the power spectral density, historically called *periodogram* [Press et al., 2007, ch.13.4, p. 652–667]. Another word for the power spectral density is *power spectrum*. Each of the transformed points corresponds to a frequency and the power spectral density is a value telling how present each frequency is in the signal. Larger values in the power spectra means higher presence of the the corresponding frequency in the signal. Distinct frequencies in the signal will be visible as peaks in the power spectrum.

The resolution of the frequencies is determined by the sampling frequency f_{samp} and the number of sampled points N and is defined as $\frac{f_{\text{samp}}}{N}$. The sampled frequencies will be integer-multiplies of the resolution. To sample periodic signals, at least *two* points needs to be sampled for each cycle for a given frequency. Given a sampling frequency f_{samp}, the maximum frequency that can be sampled will be half of that since at least two points must be sampled. The critical frequency $\frac{f_{\text{samp}}}{2}$ is called the *Nyquist frequency* [Press et al., 2007, ch. 12.1–12.2, p. 605–617]. The consequence of this is that a signal sampled every one ms has a sampling frequency of 1000 Hz, but the maximum frequency obtainable in the power spectrum is 500 Hz.

2.6.3 Averaging over time windows

The estimated power spectrum from a signal will usually contain some noise, meaning that the estimated spectrum is less smooth. One possibility to reduce the noise is to divide the duration of the total sampling time into time windows of shorter length, calculate the power spectrum for each window and average the spectra over them.

Chapter 3

Methods

3.1 NEST simulator

The studied network models are simulated using the NEST 2.14.0 software [Peyser et al., 2017]. The simulations are run with Python using the PyNEST interface. NEST is a neuronal network simulator for spiking networks with over 50 neuron models and 10 synapse models [NEST]. The simulator does not try to model the exact morphology of neurons, but rather focuses on network structure and dynamics.

There are traditionally two schemes used to update the network state in a network simulation, the *event-driven* and *time-driven* scheme. The differences in the two schemes are described by Morrison et al. [2007], Krishnan et al. [2017]. In an event-driven scheme, a central queue of events is used to determine when a neuron will fire a spike. When a neuron receives a spike, its state is updated and a prediction of when it will fire a spike is added to the central queue as an event.

The NEST simulator uses a time-driven scheme to update the neurons. In this scheme the state of the neurons are updated on a time grid. The resolution of the time grid is a simulation parameter and determines the spacing between the grid points, and thus how often the network is updated. Finer resolution results in a more accurate network simulation but at a higher computational cost. At each grid point, the neurons are checked for a threshold crossing of the membrane potential. Since the network is only updated at the grid points, the spike times are restricted to these grid points. This can cause an artificial synchrony in the network. To overcome this problem, a scheme that handles spikes in continuous time has been proposed by Morrison et al. [2007] and Hanuschkin et al. [2010]. This scheme is implemented for some model neurons in NEST and for the LIF neuron with current based synapses with exponential time cause,

the model is called `iaf_psc_exp_ps`. In the microcircuit model, external input is given by a Poisson spike train generator. A Poisson spike train is a spike train were each spike time is independent of the previous spikes. A version of the Poisson generator that enables spikes in continuous times also exists and is called `poisson_generator_ps`. The model neuron that handles spike times in continuous time will be referred to as the precise-spike-times neuron.

In the network model neurons are connected with modeled synapses. When one of the neurons in a pair of connected neurons fires a spike, the signal is received by the other neuron after a specified delay value. Each connection has its own delay parameter. Most synapse models in NEST require the delay values of the synaptic connections to be integermultiples of the time resolution. This reduces the total number of possible delay values if they are drawn from a delay distribution. This restriction limits the heterogeneity of the neurons in the network. Also implemented in NEST are synapses which can handle continuous delay values to avoid this limitation. The synapse model is called `cont_delay_synapse` and will be referred to as the continuous delay synapse.

3.2 Analysis methods

This section describes different methods used to analyse the power spectra from the simulations. To study the observed oscillating network dynamics, the power spectrum of the population averaged firing activity was used.

To obtain an estimate of the power spectrum, the time histogram for a population was obtained by using the `time_histogram` function from *Elephant.statistics*. The bin size used is 1 ms and the spike counts were averaged by the population sizes. The averaged spike counts were then transformed by the fast Fourier transformation method `numpy.fft.fft`, and the power spectrum was obtained by taking the absolute value squared of the transformed data. Finally, the power spectrum was divided by the number of sampled points to make it independent off the duration of the simulation.

3.2.1 PSD for subsamples of the populations

The firing activity of a neuron i at time t is defined by

$$s_i(t) = \lim_{\Delta t \to 0} \frac{1}{\Delta t} n_i^{\Delta t}(t) \tag{3.1}$$

where $s_i(t)$ is the firing activity of neuron i and $n_i^{\Delta t}(t)$ is the spike count of the neuron in the time interval $[t,\ t+\Delta t]$. The population averaged firing activity $s(t)$ at time t is the sum of the activity for each neuron divided by the number of neurons in the population, formally written

$$s(t) = \frac{1}{N}\sum_{i=1}^{N} s_i(t) \tag{3.2}$$

where N is the population size.

The power spectrum of the population averaged firing activity is equal to the Fourier transform of its auto-correlation and consists of two parts, the sum of the Fourier transformed auto-correlation of individual neurons and the sum of the Fourier transformed cross-correlations between all pairs of neurons [Tetzlaff et al., 2012]. The power spectrum of the population averaged firing activity is given by

$$\mathbf{C}_{SS}(\omega) = |\mathbf{S}(\omega)|^2, \tag{3.3}$$

where $\mathbf{S}(\omega)$ is the Fourier transformation of $s(t)$. The bold capital letters denotes functions in the frequency (Fourier) domain. The power spectrum can be rewritten as

$$\mathbf{C}_{SS}(\omega) = \frac{1}{N^2}\left[\sum_{i=1}^{N}\mathbf{A}_i(\omega) + \sum_{i=1,j\neq i}^{N}\mathbf{C}_{ij}(\omega)\right], \tag{3.4}$$

where $\mathbf{A}_i(\omega) = |\mathbf{S}_i(\omega)|^2$ is the power spectrum of spike train i which is the same as the transformed auto-correlation of $s_i(t)$, and $\mathbf{C}_{ij}(\omega) = \mathbf{S}_i(\omega)\mathbf{S}_j(\omega)^*$ is the cross-spectrum of spike train i and j, equal the the transformed cross-correlation of $s_i(t)$ and $s_j(t)$.

In my analysis, the number of spike trains used to calculate the power spectra is varied to see how the sample size effects the power spectra. Eq. 3.4 suggests that correlations between neurons in a populations contributes more to the compound power spectrum when the number of neurons N increases. Since the peak in the power spectrum is caused by correlations between neurons, it is expected that the peaks will be more prominent for larger subsamples, but it is unclear how many neurons is needed to be able to discern the peaks.

Data from an extended simulation time is used to increase the number of data points used in the Fourier transformation, and in that way increase the frequency resolution of the power spectra. In addition, the use of a kernel convolution on the spike counts is performed in order to get a higher time resolution of the signal before the transformation to the frequency domain.

3.2.2 Subsampling

The microcircuit model contains around 80,000 neurons, while the number of recorded neurons from experiments are only around 150 Brochier et al. [2018]. In the power spectrum of each population in the microcircuit model, both a peak at around 80 Hz and around 300 Hz are present, but the question is, how many neurons need to be recorded in order to detect the 300 Hz peak in the power spectra? To shed light on this topic, power spectra were calculated from subsets of different sizes from the microcircuit model. The spike trains used are drawn from the BDH model. All the populations in the network model were considered separately in the subsampling since the amplitude of the peaks in the power spectrum for each populations and the number of neurons is different.

The number of neurons in each population are listed in table 2.1 and ranges from 21,915 in L4E to 1,065 in L5I. In order to be able to compare the results with experimental data, subsets of not too large sizes were chosen. The subsampling was done on the populations separately, using the four different sample sizes: [50, 100, 250, 500].

The subsamples were drawn randomly from each population and with a varying number of trials. The number of trials tested was 1, 10, and 100. For each trial, the sampled spike trains were drawn randomly without replacement and the power spectrum was calculated from the time histogram of these spike trains with a bin size of 1 ms. The spike counts used to calculate the power spectra were averaged over the sample sizes to estimate the average firing activity of the subsample populations. For the subsampling with more than one trial, the resulting power spectra were averaged over all the trials.

In contrast to the power spectra calculated from the whole populations, the power spectra for the subsamples were multiplied with the subsample sizes. The reason this is done is to more naturally be able to compare the spectra from the different sample sizes. Jarvis and Mitra [2001], showed that the power spectrum for a single spike train following a homogeneous Poisson process is constant and equal to its firing rate λ. Tetzlaff et al. [2012] showed that the compound power spectrum of a population of neurons consists of the two parts, shown by Eq. (3.4). The contribution to the compound spectra from the auto-correlations of individual neurons is a sum of N terms while the contribution from the cross-correlation between pair of neurons is a sum of terms in order N^2, where N is the population size. Both of these sums are divided by the population size squared due to the averaging of the population activity. The result is that the sum of the auto-correlations of individual spike trains will contribute less to

the compound spectrum for large N compared to the sum of the cross-correlations.

Given that individual neurons in a network is well approximated by a Poisson process, then, according to Jarvis and Mitra [2001], the autocorrelation will be equal the the firing rate λ. The compound spectrum for the network will then be

$$\mathbf{C}_{SS}(\omega) = \frac{1}{N^2}\left[\sum_{i=1}^{N}\lambda + \sum_{i=1,j\neq i}^{N}\mathbf{C}_{ij}(\omega)\right] \tag{3.5}$$

$$= \frac{1}{N^2}\sum_{i=1}^{N}\lambda + \frac{1}{N^2}\sum_{i=1,j\neq i}^{N}\mathbf{C}_{ij}(\omega). \tag{3.6}$$

When the neurons are uncorrelated, the cross-correlation will be zero and the latter term will vanish, giving the following equation for the power spectrum

$$\mathbf{C}_{SS}(\omega) = \frac{1}{N^2}\sum_{i=1}^{N}\lambda = \frac{\lambda}{N}. \tag{3.7}$$

For frequencies where the neurons are uncorrelated and individual neurons fires irregularly, the power spectrum is approximated by the above equation. The division of N means that the power spectrum for uncorrelated frequencies will be lower when N increases. The power spectra for the subsampling is multiplied by N in order to avoid these vertical shifts.

For frequencies where the neurons are correlated, the power spectrum will get a contribution from the som of the cross-correlations between neurons, resulting in increased power. This corresponds to the observed peaks.

3.2.3 Extended simulation time

Individual neurons in the BDH model has an average firing rate of a few Hz, depending on the population (see Fig. 1D from Bos et al. [2016]). The reduced number of neurons means that more bins in the time histogram will contain zero count compared to the full network. All the simulations used up to this point have been run for 10 seconds, and the power spectra have been calculated from the last 9 seconds of those. To see whether the duration of the simulation has an effect on the detectability of the observed peaks in the power spectra of the subsamples, the spectra was calculated from the simulation time between 1s–100s with the same BDH model using a time histogram width a bin size of 1 ms.

3.2.4 PSD from higher resolution firing activity

As an alternative to using the time histogram directly to calculate the power spectrum, a kernel convolution can be performed on the spike counts. A kernel convolution of the spike counts is another way to get the firing activity in a network, but the method avoids the effects of binning.

Mathematically a convolution $h(x)$ of two functions $f(x)$ and $g(x)$ is defined as [Kreyszig, 2011]:

$$h(x) = (f * g)(x) = \int_{-\infty}^{\infty} f(t)g(x-t)dt \tag{3.8}$$

Here, two types of kernels are presented, a rectangular kernel and a Gaussian kernel. Both of these kernels have a parameter σ determining the bandwidth of the kernel Elephant Kernels. The size of the rectangular kernel is defined as

$$K(t) = \begin{cases} \frac{1}{2\tau}, & |t| < \tau \\ 0, & |t| \geq \tau, \end{cases} \tag{3.9}$$

where $\tau = \sqrt{3}\sigma$ and the Gaussian kernel is defined as

$$K(t) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{t^2}{2\sigma^2}} \tag{3.10}$$

Both of these kernels are defined such that the variance of the kernel function is σ^2. In other words, it satisfies:

$$\int_{-\infty}^{\infty} t^2 K(t)dt = \sigma^2 \tag{3.11}$$

The kernels used was a rectangular kernel with $\sigma = 0.2$ and a Gaussian kernel with $\sigma = 0.2$. Both kernels were loaded from the *Elephant.kernels* module. The kernel convolution was done by first getting the spike counts from a time histogram with bin size of 0.1 ms. The number was chosen equal to the time resolution of the simulation. Then the convolutions were performed with the *fftconvolve* function from *scipy.signal*.

The output after the kernel convolution is an estimate of the firing activity, but obtained with a finer binning yielding a higher sampling frequency compared to the coarser binning. The convoluted activity is transformed with the FFT algorithm and the psd is calculated the same way.

3.3 Analytical power spectra

The following section presents the theory used by Bos et al. [2016] to estimate the analytical power spectra for the population averaged firing rate. The method relies on the fact that the dynamics of a network of LIF neurons, such as those used in the microcircuit model, is well approximated by a linear rate model [Grytskyy et al., 2013]. The observed firing activity of the rate model is described by a fluctuating with output noise.

$$y_i(t) = r_i(t) + x_i(t) \tag{3.12}$$

where for population i y_i is the observed rate, r_i is the fluctuating rate and x_i is the noise modeled as Gaussian white noise.

To describe the fluctuating firing activity $r(t)$ the stationary state of the populations is first estimated using mean field theory. The stationary state is characterized by the mean firing rate of the populations. In the next step, linear response theory is applied to model the response of incoming spikes and used to describe fluctuation around the stationary state. From the correlation of the observed population activity, the power spectra is calculated.

It is important to note that the analytical results are only applicable for networks in the AI regime and not SI regime. The theory behind the analytical results is based on the fact that the total input to a cell can be approximated as Gaussian white noise, given sufficiently asynchronous activity [Grytskyy et al., 2013].

To get the stationary state of the network, the mean firing rates of the populations are estimated by solving the Fokker-Planck equation for the probability distribution for the membrane potentials. This formalism comes from Fourcaud and Brunel [2002]. The equation of the firing rate for population i consisting of LIF neurons with exponentially decaying current is taken from Eq. (6.42) by Schmidt [2016] and is given by

$$\frac{1}{\bar{r}_i} = \tau_r + \tau_m \sqrt{\pi} \int_{\frac{V_r-\mu_i}{\sigma_i}+\gamma\sqrt{\frac{\tau_s}{\tau_m}}}^{\frac{\theta-\mu_i}{\sigma_i}+\gamma\sqrt{\frac{\tau_s}{\tau_m}}} [1+\mathrm{erf}(x)]\, e^{x^2} dx, \tag{3.13}$$

where $\bar{r}_i$ is the mean firing rate, τ_r is the refractory period, τ_m is the membrane time constant, V_r is the resting membrane potential, θ is the threshold value of the potential and $\mathrm{erf}(x)$ is the error function [DLMF, Eq. 7.2.1]. μ_i and σ_i^2 are defined as

$$\mu_i = \tau_m w \left(\sum_{j \in E} K_{ij} \bar{r}_j - g \sum_{j \in I} K_{ij} \bar{r}_j + K_{\text{ext},i} r_{\text{ext}} \right) \tag{3.14}$$

$$\sigma_i^2 = \tau_m w^2 \left(\sum_{j \in E} K_{ij} \bar{r}_j - g^2 \sum_{j \in I} K_{ij} \bar{r}_j + K_{\text{ext},i} r_{\text{ext}} \right). \tag{3.15}$$

where w is the synaptic weight, K_{ij} is the number of connections from population j to population i and $K_{\text{ext},i}$ and r_{ext} is the number of external connections and the firing rate of the external signal respectively. Note that the synaptic delays are not included in the equation and thus do not effect the stationary state of the network.

The formula describing the fluctuating activity is given by Eq. (11) in Bos et al. [2016] and reads.

$$r_i(t) = \int_{-\infty}^{t} \sum_{j=1}^{N} M_{ij}^A \, H_{ij}(t-s) \left(r_j(s-d_{ij}) + x_j(s-d_{ij}) \right) \mathrm{d}s, \tag{3.16}$$

where $H_{ij}(t)$ is the impulse response of a neuron receiving a spike at time t and d_{ij} is the delay of a signal from population j to population i. M_{ij}^A is an entry of the connectivity matrix M^A and describes the strength of the connection from population j to population i. N is here the number of populations. The matrix is defined as is the element-wise product of the number of connections and synaptic weights from the anatomical connectivity map of the microcircuit model. Applying a Fourier transformation on the equation replaces the convolution by a multiplication and makes it easier to solve. By doing the substitution of the variable $s \to s + d_{ij}$ in Eq. (3.16) the equation becomes

$$r_i(t) = \int_{-\infty}^{t} \sum_{j=1}^{N} M_{ij}^A \, H_{ij}(t-s-d) \left(r_j(s) + x_j(s) \right) ds. \tag{3.17}$$

The firing activity r_j and the noise x_j are independent of the delays. One can now define a new matrix called the *effective connectivity matrix* which is the element-wise product of M^A and $H(t-s-d)$. This matrix is denoted $M_d(t-s-d)$. The subscript d is to emphasize the dependency of the delays.

The observed rate fluctuation can be found by transforming Eq. (3.17) in the Fourier space and yields

$$\mathbf{R}(\omega) = \mathbf{M}_d(\omega)(\mathbf{R}(\omega) + \mathbf{X}(\omega)) \tag{3.18}$$

$$\tag{3.19}$$

Here, the bold letters denotes the variables in Fourier space. The firing activity is a function of ω, which is related to the frequency by $\omega = 2\pi f$.

Given the definition of the rate model in Eq. (3.12), the fluctuating activity of the network is given by

$$\mathbf{Y}(\omega) = \mathbf{R}(\omega) + \mathbf{X}(\omega) \tag{3.20}$$

The power spectrum for one population is defined as the auto-correlation of the network activity for that population. Therefore, the power spectra for all the populations can be read in the diagonal of the correlation matrix of $\mathbf{Y}$ defined as

$$\mathbf{C}(\omega) = \left\langle \mathbf{Y}(\omega)\mathbf{Y}^T(-\omega) \right\rangle . \tag{3.21}$$

3.3.1 Description of the transfer function

The impulse response H_{ij} is also called the transfer function and describes how the firing activity of one population reacts to am input signal. The derivation of the transfer function of a LIF neuron driven by colored noise is given by [Schuecker et al., 2015, Eq. 30]. The parabolic cylinder functions is part of the equation where the solutions can be found in several standard libraries such as *scipy* [Jones et al., 2001–]. In the supplementary material to Bos et al. [2016], a Fortran implementation of the parabolic cylinder functions was used. However, all the necessary files to get it to work were not included in the material. Therefore, the scripts found at `https://github.com/INM-6/neural_network_meanfield` was used instead. The solution of the parabolic sylinder functions comes from the *mpmath* package in Python Johansson et al. [2013], which is slower than the Fortran implementation but sufficed for the purpose in this thesis. The transfer function returns an 8×8 matrix for each ω and the stack of each matrix corresponding to frequencies from 0 Hz to 500 Hz was calculated once and then saved. The relationship between ω and frequency f is $\omega = 2\pi f$, meaning that the spacing between the frequencies is $\frac{1}{2\pi}$. The theoretical mean firing rates were also calculated once and saved. The results didn't need to be recomputed for each change in the model since the transfer function does not depend on the delays which was the only parameter that was changed.

3.3.2 Contribution of synaptic delays

In the equation for the observed rate fluctuations 3.18, the delays enter the model only through the effective connectivity matrix $\mathbf{M}_d$, and in Fourier space it is possible to separate this the delay term from the other terms as

$$\mathbf{M}_{d,ij}(\omega) = \mathbf{M}_{ij}(\omega)e^{-i\omega d_{ij}}, \tag{3.22}$$

where the delay term $e^{-i\omega d_{ij}}$ is the average over all possible values for the delay d_{ij}. This averaging is done by an integration over all the the possible values weighted by the probability density function (pdf) of the delay distribution, mathematically described by

$$e^{-i\omega d_{ij}} = \int_{-\infty}^{\infty} e^{-i\omega y} p(y)\, \mathrm{d}y, \tag{3.23}$$

where $p(y)$ is the probability density function for the delays. The effective connectivity matrix can in this way be defined for different probability distributions for the delays as shown in the next section.

3.4 Delay distributions

In the microcircuit model, the delays are drawn randomly from a truncated Gaussian distribution with parameters specified in Table 2.3, and with the delay standard deviation doubled for the BDH model. This delay distribution of the original microcircuit model is truncated at 0 because the delays need to be positive. To determine the contribution of the delay distribution to the effective connectivity matrix for the different delay distributions, the averaging defined above needs to be performed for the *pdf* of the distribution. Bos et al. [2016] show the result for the original truncated Gaussian distribution. In this thesis, power spectra obtained for three more delay distributions; the exponential, lognormal and uniform distributions. After calculating the contribution for the effective connectivity matrix for these distributions the analytical spectra can be calculated with the same procedure as Bos et al. [2016].

3.4.1 Truncated Gaussian

The effective connectivity matrix with the truncated Gaussian distribution is given in Bos et al. [2016] and shown here for completeness. The pdf is defined as:

$$p(x;\mu,\sigma) = \frac{1}{\sqrt{2\pi}\sigma\left(1-\Phi\left(\frac{-\mu}{\sigma}\right)\right)}\int_0^\infty e^{-\frac{(x-\mu)^2}{2\sigma^2}}dx, \tag{3.24}$$

where $\Phi(x)$ is

$$\Phi(x) = \frac{1}{2}\left[1+\text{erf}\left(\frac{x}{\sqrt{2}}\right)\right], \tag{3.25}$$

and $\text{erf}(x)$ the error function DLMF defined as

$$\text{erf}(x) = \frac{2}{\sqrt{\pi}}\int_0^x e^{-t^2}dt \tag{3.26}$$

The effective connectivity matrix for this distribution after the averaging is given as

$$\mathbf{M}_{d,ij}(\omega) = \mathbf{M}_{ij}(\omega)\frac{1-\Phi\left(\frac{-\mu_{ij}+i\omega\sigma_{ij}^2}{\sigma_{ij}}\right)}{1-\Phi\left(\frac{-\mu_{ij}}{\sigma_{ij}}\right)}e^{-i\omega\mu_{ij}}e^{-\frac{\sigma_{ij}^2\omega^2}{2}} \tag{3.27}$$

3.4.2 Exponential distribution

The pdf for the exponential distribution is given by

$$p(x;\lambda) = \lambda e^{-\lambda x} \tag{3.28}$$

The averaging over all possible delay values yields

$$\int_0^\infty e^{-i\omega x}\lambda e^{-\lambda x}dx = \lambda\int_0^\infty e^{-(i\omega+\lambda)x}dx \tag{3.29}$$

$$= -\lambda\frac{1}{i\omega+\lambda}\left[e^{-(i\omega+\lambda)x}\right]_{x=0}^{x=\infty} \tag{3.30}$$

$$= \frac{\lambda}{i\omega+\lambda} \quad \text{if } \omega,\lambda \geq 0 \tag{3.31}$$

which yields the expression for element in the effective connectivity matrix

$$\mathbf{M}_{d,ij}(\omega) = \mathbf{M}_{ij}(\omega)\frac{\lambda}{i\omega+\lambda}. \tag{3.32}$$

3.4.3 Lognormal distribution

The pdf for the lognormal distribution is

$$p(x|\mu\sigma) = \frac{1}{x\sigma\sqrt{2\pi}} e^{-\frac{(\ln(x)-\mu)^2}{2\sigma^2}} dx \tag{3.33}$$

The averaging over all possible delay values is

$$\mathbf{M}_{d,ij}(\omega) = \int_0^\infty e^{-i\omega x} \frac{\mathbf{M}_{ij}}{x\sigma\sqrt{2\pi}} e^{-\frac{(\ln(x)-\mu)^2}{2\sigma^2}} dx \tag{3.34}$$

There is no obvious way to get a nice expression for the averaging over all delay values for this distribution. The averaging was therefore done by numerical integration of Eq. (3.34). Note that the only part of the equation that is complex is the $e^{-i\omega x}$ term. By using Euler's identity $e^{-i\omega x} = \cos\omega x - i\sin\omega x$ one can separate the real and complex part of the equation into two terms and integrate each of them separately. The numerical integration was done using the function `simps` from scipy.integrate on on the time segment from 0.1 to 100 ms with a step size of 0.0001 ms. The cumulative probability of the distribution from 0 to 100 ms was found to be sufficiently close to one, so this step size and duration ensured that the numerical error was small. The integration from 0.1 ms was chosen because all delay values in a simulation with NEST cannot be smaller than the time resolution which was set to 0.1 ms in the simulations performed. To be exact, the log-normal distribution should be renormalized to account for the truncation at 0.1 ms, but the probability that a delay value lies in the interval [0, 0.1] was found to be around 0.01. Because of this low probability, the renormalization was ignored to simplify the expression. To save some computational time, the step size and duration could be chosen more optimally, but the numerical calculations was only computed once and therefore this optimization was not prioritized.

3.4.4 Uniform distribution

The pdf for the uniform distribution is defined by

$$p(x;a,b) = \frac{1}{b-a}, \tag{3.35}$$

where a and b is start and end time of the uniform distribution.

The averaging over all possible delay values is

$$\int_a^b e^{-i\omega x}\frac{1}{b-a}dx = -\frac{1}{(b-a)i\omega}\left[e^{-i\omega x}\right]_{x=a}^{x=b} \tag{3.36}$$

$$= \frac{1}{(b-a)i\omega}\left(e^{-i\omega a} - e^{-i\omega b}\right) \tag{3.37}$$

An element of the effective connectivity matrix is then

$$\mathbf{M}_{d,ij} = \frac{\mathbf{M}_{ij}}{(b-a)i\omega}\left(e^{-i\omega a} - e^{-i\omega b}\right) \tag{3.38}$$

3.4.5 Using the different delay distributions

It was been shown by Bos et al. [2016] that the connections between the four inhibitory populations to themselves are most sensitive to changes in the effective connectivity matrix. Therefore, the delay distributions are only changed for these four connections, leaving all the other connections unchanged. The four connections can be found in the connectivity matrix as the entries M_{22}, M_{44}, M_{66} and M_{88}. When referring to the model with exponential delay distributions for instance, it is implied that only these four inhibitory connections use the exponential distribution for the delays. The analytical spectra are calculated for all populations for each delay distribution.

3.5 Trajectory of eigenvalues

To better understand the analytical spectra, the trajectory of the eight eigenvalues from the effective connectivity matrix were plotted in the complex plane parameterized by the frequency. These eigenvalues determines different modes of the network and the observed power spectra is a combination of these *eigenmodes*. By the use of eigenvalue decomposition of the effective connectivity matrix, it can be shown that the analytical power spectra can be described as a sum of terms containing by $|1/(1-\lambda_i(\omega))|$, where λ_i is the i$^{\text{th}}$ eigenvalue. When the i$^{\text{th}}$ eigenvalue approaches the point 1 that eigenmode starts to dominate the spectra and results in large power. The contributions of the eigenmodes when the eigenvalues approaches 1 results in the observed peaks in the power spectra.

3.6 Parameter sweep

This section presents the method used to study the effects on the peaks in the analytical power spectra of using different combinations of the mean and the standard deviation of the truncated Gaussian distribution. This was done in order to understand more of how the oscillating dynamics in the network depend on the delays. The change of the delay parameters was done on *all* the inhibitory connections. The reason why all the inhibitory connections was changed and not only the four inhibitory connections used in section 3.4.5 was to limit the model to only use two different delay distributions for all the neurons. One for the excitatory and one for the inhibitory neurons.

3.6.1 High frequency peak

Bos et al. [2016] found that a loop in the connections from the four inhibitory populations to themselves generates the high frequency peaks. It was found that the frequency of the observed high frequency peak in the power spectra was close to the frequency where the entries $\mathbf{M}_{22}$, $\mathbf{M}_{44}$, $\mathbf{M}_{66}$ and $\mathbf{M}_{88}$ in the effective connectivity matrix $\mathbf{M}_d$ was closest to the point 1.

The first step was to locate this frequency as a starting point. Finding the point in $\mathbf{M}_d$ closest to 1 for all the four connections mentioned above resulted in roughly the frequency. Only one of them was therefore needed. The second step was to calculate the spectrum for one of the eight populations, start at the estimated frequency and move in the direction of increasing power until a local maximum was found, or the frequency was either 0 HZ or 500 Hz (the boundary points).

This procedure was done for all combinations of the means in the range (0.2–4.0) ms and standard deviation in the range (0.2–3.0) ms with step size of 0.1 ms. For each combination the mean and standard deviation, the amplitude and the frequency of the peak were calculated and stored in two 29×39 matrices. The entry i, j in the matrices corresponding to the results using standard deviation number i and mean number j.

This method aims at locating the high frequency peaks and will therefore be referred to as the high-frequency method in this context.

3.6.2 Low frequency peak

The frequency of the low frequency peak is easier to find. It was observed that the peak always was located at frequencies above 30 Hz for the parameters of mentioned above. Therefore, starting at 30 Hz the frequency was increased until a local maxima

was found in the power spectrum. The frequency and the amplitude of the peak were stored in data matrices as for the high frequency peak. This method finds the peak in the power spectrum with the lowest frequency and will therefore be referred to as the the low-frequency method.

3.7 Multi-area model

The following section briefly describe the multi-area model [Schmidt et al., 2018a,b, Schuecker et al., 2017].

The multi-area model is a neuronal network model that combines several microcircuit models, all representing different areas in the brain. In addition to the connections between the populations, there are additional connections across the different areas as well. The inclusion of more microcircuit models and connections results in a much larger network. The network model was run on the Piz Daint computer in Switzerland https://www.cscs.ch/computers/piz-daint/. The model was simulated using 16 nodes each handling 36 virtual processes. The model can be found on https://github.com/INM-6/multi-area-model, where the parameters used in the model can be found in the `default_params.py` file. In contrast to the microcircuit model, two new parameters χ and χ_I were included to scale the weight of the synaptic connections across different areas. Synaptic connections from other areas onto excitatory neurons are scaled by the factor χ and the synaptic connections from other areas onto inhibitory neurons are scaled by $\chi\chi_I$. For simulations used in my analysis, the parameter χ and χ_I determining the strength of the connections across different layers to the excitatory and inhibitory populations were altered. The first simulation used $\chi = 1$ and $\chi_I = 1$ and the second simulation used $\chi = 1.9$ and $\chi_I = 2.0$. The reason why two different models were run was to see if the stripes occur in the network for any of the strengt values. The power spectrum from population L4E of the V1 area in the multi-area models was computed and compared with the spectra for L4E from the original BDH model and the BDH model with exponential delays for the four connections from the inhibitory connections to themselves.

3.8 Description of the simulations

The microcircuit model were simulated using the NEST simulator 2.14. [Peyser et al., 2017] software. All the models were run for a duration of 10 seconds with time res-

olution 0.1 ms, except for one model that was run for 100 seconds. The script for the network model of Potjans-Diesmann was modified to incorporate the changes explained in Bos et al. [2016] and to include precise spike times neurons, continuously delays, and other delay distributions described in section 3.1. The script was modified to be able to accept a parameter file with specifications of the wanted modifications. This allowed the simulations to be run in parallell on the Stallo computer cluster without making more than one modified microcircuit model. Stallo is a computer cluster own by UiT The Arctic University of Norway. The computer cluster runs scripts with the requested number of CPU's and CPU cores. All the microcircuit simulations were separated over 4 CPU cores except the 100 seconds long simulation which was separated over 8 CPU cores.

In all the calculations of the power spectra, the first second was skipped to avoid effects of the initialization of the simulations. Unless otherwise stated, the power spectra was calculated from a time histogram using a bin width of 1 ms.

3.9 Python libraies

An overview of all the module versions is listed in Table 3.1.

3.9.1 Tools for handling neurophysiological data

Neo is a tool that handles different electrophysiological data i Python Garcia et al. [2014]. This tool provides an easy way to load the data and extract the spike trains from the simulation results. The library elephant (Electrophysiology Analysis Toolkit) is a tool for analyzing electrophysiological data [Elephant]. Together with Neo, these two packages made the data management convenient.

Table 3.1: Python packages used in this thesis

Name	Version	Name	Version	Name	Version
Python	3.6.8	Matplotlib	2.2.3	SciPy	1.1.0
NumPy	1.15.4	quantities	0.12.2	NEST	2.14.0
Elephant	0.6.0	pandas	0.23.4	mpmath	1.0.0
Neo	0.7.1	seaborn	0.9.0		

Chapter 4

Results

This chapter presents the results from the analysis of the vertical stripes. The first part shows the relation between the peaks in the power spectrum and the vertical stripes in the raster plot, and that the power spectra obtained by time binning of 1 ms and kernel convolution yields similar results.

The next part compares the PD model and BDH model and shows how the different neuron and synapse models affects the dynamics of the network. Power spectra and raster plots will be used for visualization. The part after that presents the result of the subsampling and how detectable the peaks in the power spectra are. Then, inspired by Bos et al. [2016] analytical results of the power spectra were analyzed to see the effect of using different delay distributions and delay parameters. The final part shows results from simulations of the multi-area model in comparison with other microcircuit models.

4.1 Verifying the analysis methods

This section connects the observed vertical stripes with the peaks in the power spectra and compares the results from the two power spectrum calculation methods.

From a raster plot one gets an idea about the dynamics of the network, and can see whether there are possible correlations in the spiking activity or not. The vertical stripes analyzed in this thesis are one such sign of correlated activity. The analysis done in Bos et al. [2016] suggests that the observed vertical stripes are visible in the power spectrum of the population averaged firing activity. To test this claim, a small jittering of each spike time, with values drawn from a Gaussian distribution with mean 0 and standard deviation 1 ms, was performed to smooth the stripes of the fast os-

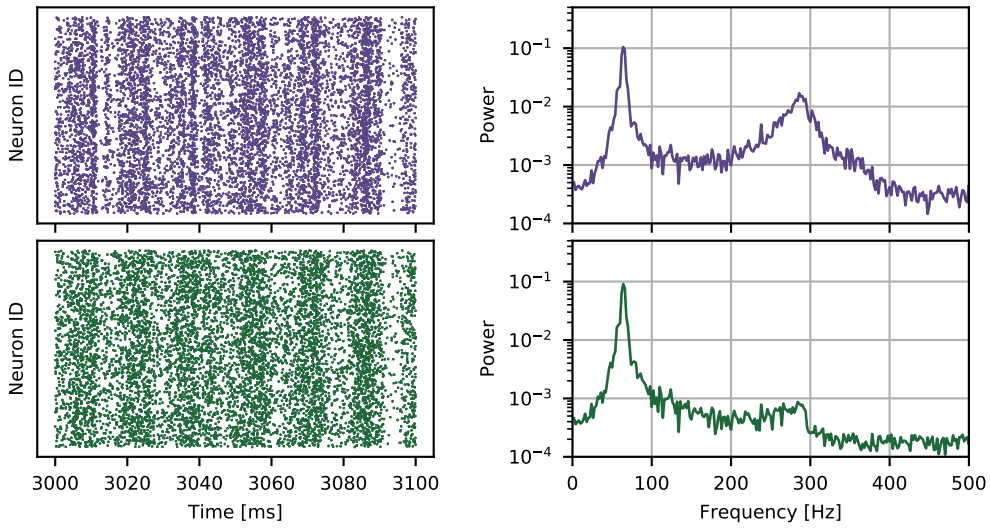


Figure 4.1: Comparison of jittered spike times (green) and non-jittered (purple) spike times for layer L4E from the BDH model. The raster plots on the left shows 100 ms of simulation time and the power spectra are calculated from the last 9 s of the total simulation time and averaged over time windows with size 500 ms. Each individual spike time is shifted by an amount drawn from a Gaussian distribution with mean 0 ms and standard deviation 1 ms.

cillations. Then the power spectrum was calculated to and compared with the power spectrum for the non-jittered spike times. This method was applied to the population L4E only since the peak in the power spectrum for this population is the most prominent. Fig. 4.1 shows the comparison of jittered vs non-jittered spike times. As can be seen in the figure, both the thin vertical stripes and the high frequency peak are almost gone after the jittering, which suggests that the two aspects are related. Note that the low frequency peak in the power spectrum is not affected by the small jittering. This peak most likely represents the wider stripes still visible in the raster plots.

Since there is reason to believe that the peaks in the power spectra and the vertical stripes are related, changes in the stripes will be studied by looking at the population averaged power spectra. The frequency of the peaks corresponds to the frequency of the oscillation in the raster plot, while the amplitude of the peak says something about how prominent the stripes are.

The population averaged power spectra for each population in the microcircuit model can be calculated by two methods as explained in section 3.2 and section 3.2.4. The power spectra from the kernel convolution was calculated for population L4E of the BDH model to compare the spectra obtained from the two methods. The results are

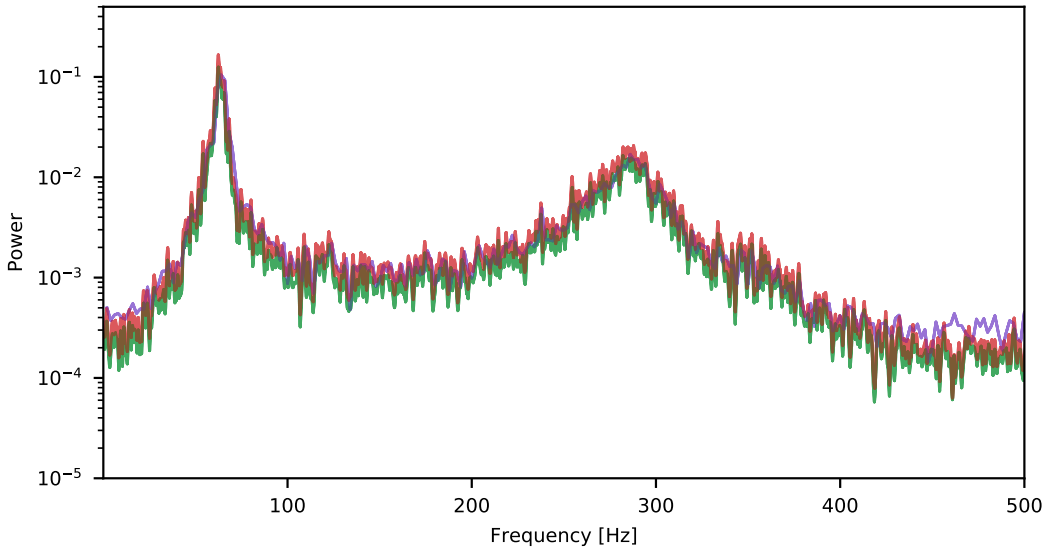


Figure 4.2: Power spectra from population L4E from the BDH-model calculated using time histogram and a kernel convolution with a rectangular and a Gaussian kernel. Purple: power spectrum using a time histogram with bin size 1 ms and averaged over 500 ms windows. Green: power spectrum calculate from the kernel convolution with rectangular kernel. Red: power spectra calculated from the kernel convolution with a Gaussian kernel. Both kernels use parameter $\sigma = 0.2$.

shown in Fig. 4.2. Both the use of a rectangular and a Gaussian kernel yielded almost the exact same power spectra. The only difference is a slight difference in amplitude, but the shape is the same. For higher frequencies the amplitude of the power decreases for the kernel convolution. That is an effect of the kernel size used. Larger band-width of the kernel tends to smooth out the spike counts for shorter lengths and makes higher frequencies invisible in the power spectra. As seen in the figure, the effect is not visible for frequencies under 400 Hz and therefore does not affects the peaks.

The use of kernel convolution yielded similar results as the time binning, and verifies that the use of time binning as an estimate of the power spectra is sufficient. Therefore, the time binning with a bin size of 1 ms will be used to calculate the power spectra for the rest of the analysis.

4.2 Different neuron and synapse models

This section first shows a comparison of the dynamics of the PD and BDH models. Then, the dynamics of the microcircuit models with different neuron and synapse models are compared to see the effects on the oscillatory activity.

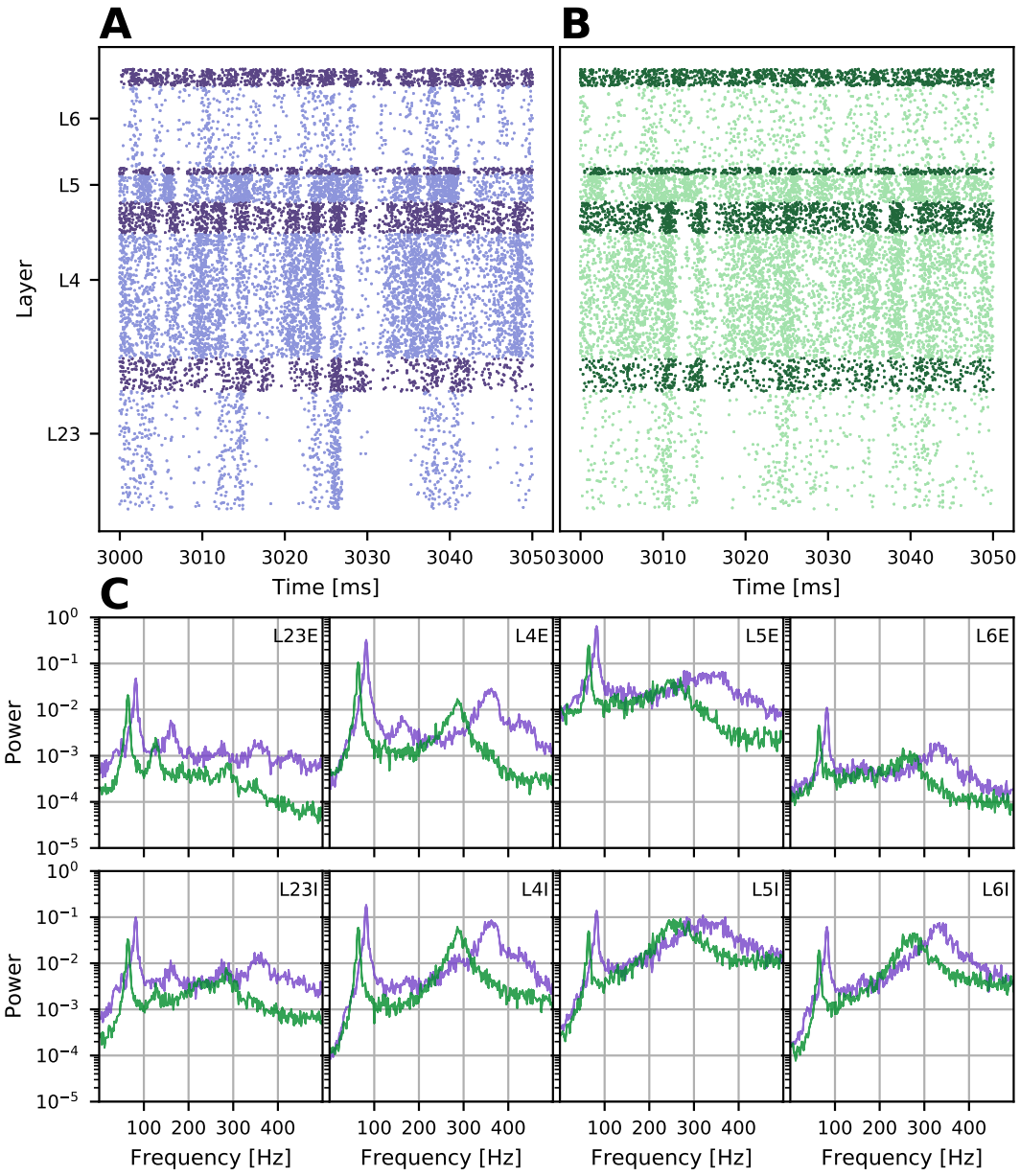


Figure 4.3: Comparison of the PD model (purple) and BDH model (green). The raster plots shows all the eight populations were the dark color represents the inhibitory populations and the light color the excitatory populations.

4.2.1 Comparison of the microcircuit models

Two version of the microcircuit model have been presented up to this point, the original PD model and the BDH model. The power spectra and raster plot for the two models are shown in Fig. 4.3. For the BDH model, both the peaks are shifted to the right, while the high frequency peaks are shifted the most. There are also some differences in the amplitude. The vertical stripes are still present for both models in the raster plots.

The us of different neuron and synapse models was tested for both the PD model and the BDH model. Since the effect of changing the neuron and synapse models on the dynamics was the same for both network models, only the result from the BDH model is shown here. The BDH model was chosen since it was showed to be in a state farter from the SI regime [Bos et al., 2016].

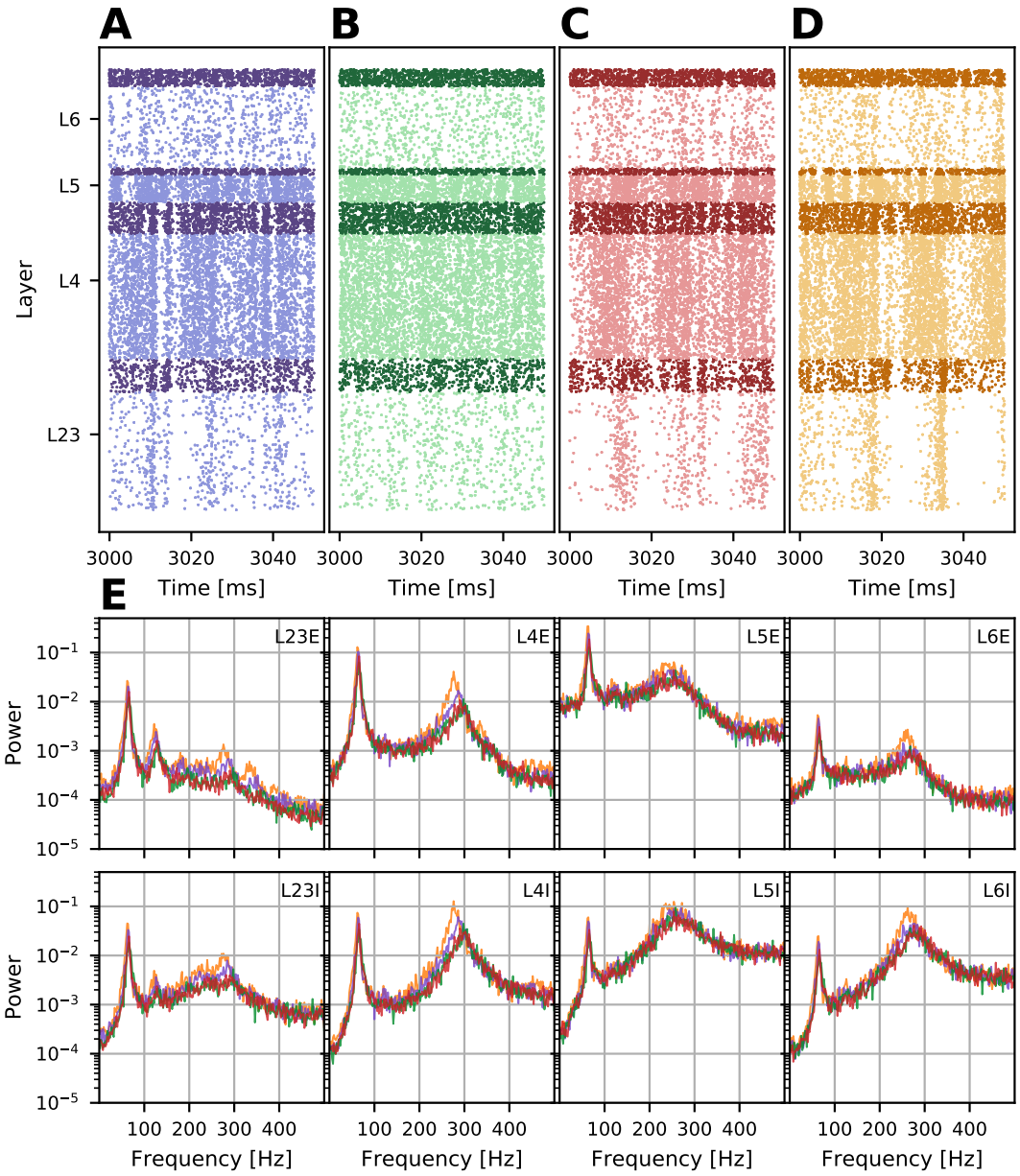


Figure 4.4: Power spectra from the BDH model. (A–D): Raster plot from 50 ms of the simulation. The dark colors shows the inhibitory and the light colors the excitatory populations. **A**: Original model, **B**: model with precise-spike-time neurons, **C**: model with both precise-spike-time neurons and continuous delays and **D** continuous delays but normal normal spike times. **E**: Power spectra for the 8 different populations. The top row shows the spectra of the excitatory populations. The bottom row shows the spectra from the inhibitory populations. The colors of the spectrum lines matche the colors of the models in A–D. The power spectra are calculated using a time histogram with 1 ms bin width and is averaged over time windows of 500 ms.

4.2.2 Using different neuron and synapse models

The first part of the analysis is to see how the use of precise spike time neurons and continuous delay synapses affects the observed stripes. Four different network models were simulated. One with the original neuron and synapse models, one with precise-spike-times neurons, one with continuous delays synapses and one with both precise-spike-times neurons and continuous delays.

Fig. 4.4 shows the results. As can be seen in the figure, both the vertical stripes and the peaks in the power spectra are still present. The use of precise-spike-times neurons shifts the power spectra a tiny amount to the right, but the stripes are still visible in the raster plots. The use of precise-spike-times neurons and precise-spike-times neurons with continuous delays shown in green and red yields almost identical power spectra. The yellow line, representing the model with continuous delays *without* precise-spike-times neurons, is shifted slightly to the right compared to the original BDH model.

For the low frequency peak, there seems to be almost no difference between the models. Fig. 4.5 shows the low frequency peak for population L4E for all the models and the vertical stripes are still visible in the raster plots.

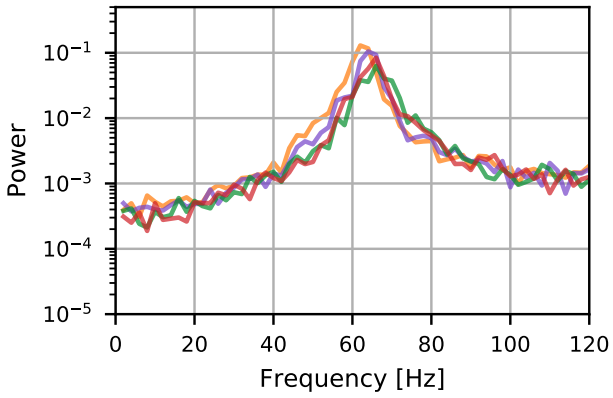


Figure 4.5: Inset of the low frequency peak for population L4E from the simulations of the BDH model with different neuron and synapse models. Purple: Original BDH model. Green: Precise-spike-times neurons. Red: precise-spike-times neurons and continuous delays. Yellow: Continuous delays. The power sepctra are calculated as in Fig. 4.4.

4.3 Subsampling

With few spike trains included, the total number of spikes at each time bin in a time histogram will be considerably lower, as illustrated in Fig. 4.6. The figure shows the time histograms and raster plots for population L4E for subsamples with 50, 100, 250 and 500 number of neurons and the whole population. The time histograms have a bin size of 1 ms and the raster plots shows 200 ms of the simulation time. For a sample size of only 50 spike trains, the spike counts in the time histogram are mostly 0's and 1's. The variety in the spike counts increases as the number of neurons in the subsample gets larger. The fewer neurons in the raster plots makes it harder to determine whether the vertical stripes are present or not.

The power spectra from the subsampling are shown in Fig. 4.7. The spectra are calculated for all 8 populations each with sample sizes of 50, 100 250 and 500 number of neurons. The subsampling was done for different number of trials, each shown by different colors and separate subplots. Each of the power spectra was multiplied by its sample size in order to get the spectra on the same scale as explained in section 3.2.2.

First, we see that the peaks in the power spectra are clearly visible even though the vertical stripes in the raster plot in Fig. 4.6 are harder to make out. Both the low and the high frequency peak becomes more visible in the power spectra for larger sample sizes. The average of an increased number of trials reduces noise in the power spectra and thus makes the peak easier to discern for smaller sample sizes. The increase in the number of trials does not result in higher peaks in the spectra.

In the original power spectra for each population shown in Fig. 4.4, the peak at around 300 Hz is more visible for some populations than other, and most prominent in population L4I. The difference in the amplitude is somewhat reflected in the subsampling as well. For population L4I, the peak is slightly visible from a subsampling of only 50 neurons after averaging over 10 or 100 trials. For population L23E on the other hand, the spectrum is flat at 300 Hz with the same sample size.

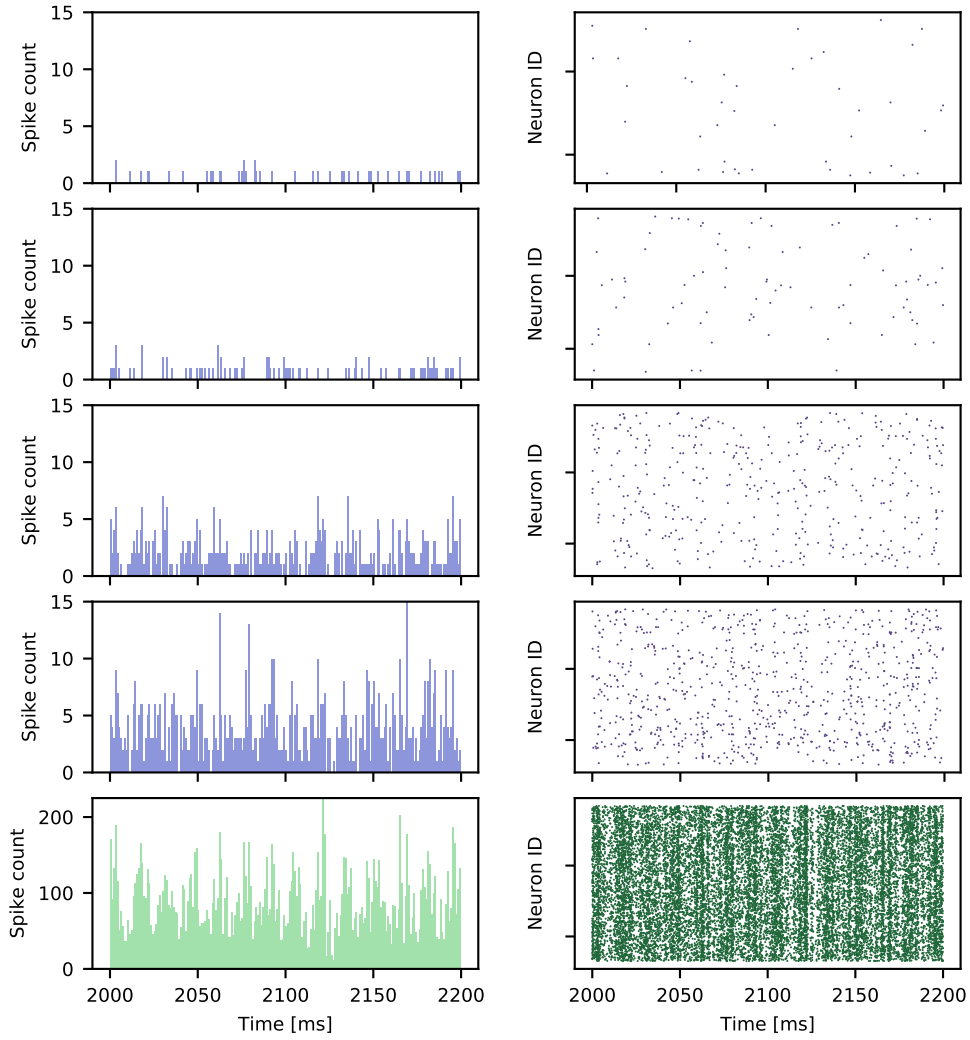


Figure 4.6: Time histograms and raster plots from subsampling of population L4E of the BDH-model. The purple plots show from top to bottom the results for sample sizes 50, 100, 250, 500. The green plots at the bottom show the results for the whole population. Note that the y-axis in the time histogram for the full model size is different than that for the subsamples in order to show the full height of the bars. The bin width of the time histograms is 1 ms.

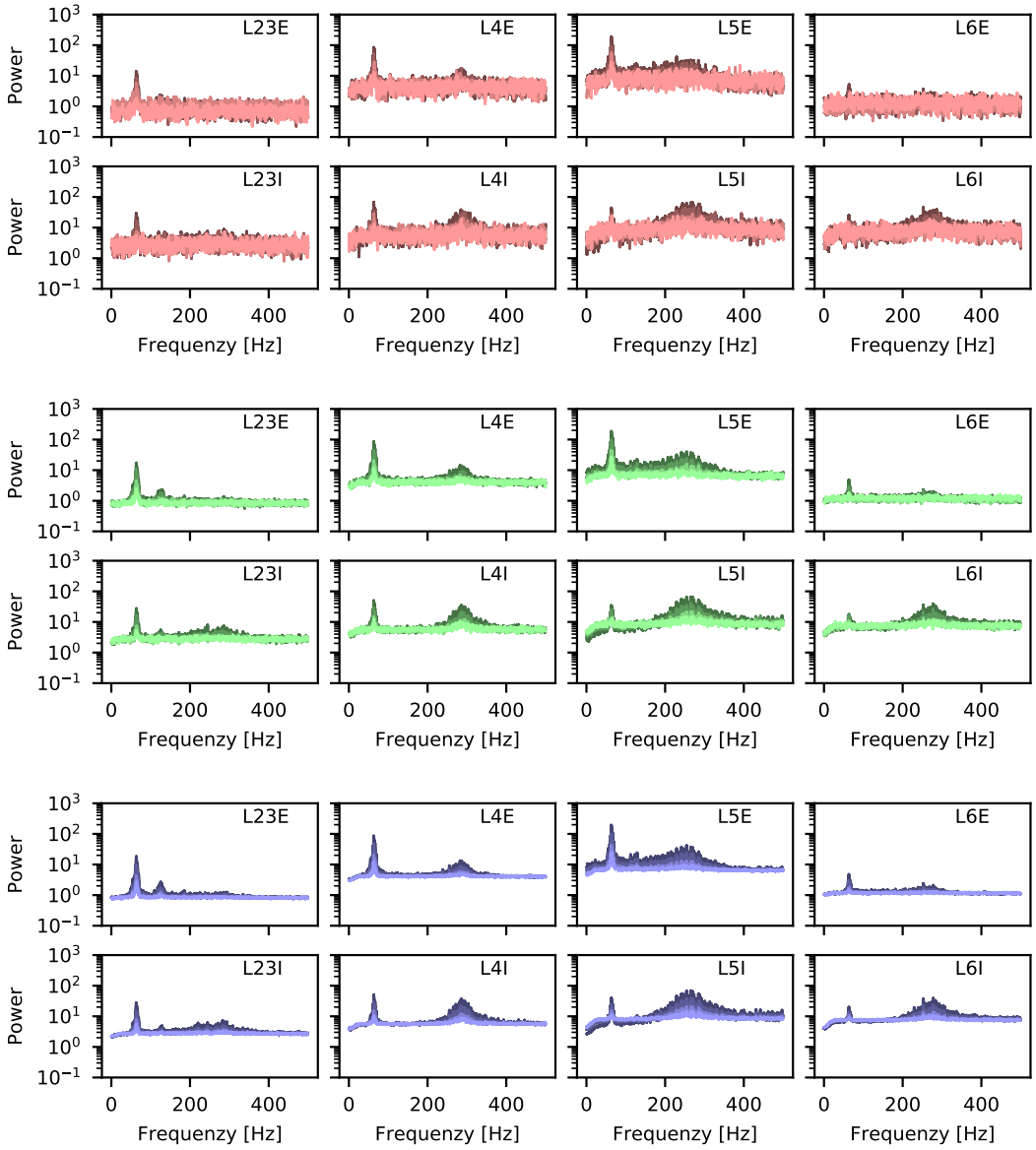


Figure 4.7: Subsampling of the spike trains from each of the populations in the BDH model. Spectra calculated from the final 9 seconds of the simulation. The spectra for the different sample sizes are multiplied with its sample size to plot each spectra on the same scale. Red color shows the spectra from 1 trial, green shows average spectra of 10 trials and purple shows average spectra from 100 trials. The brightness of the colors indicated the four different sample sizes, from light to dark colors the sample sizes are 50, 100, 250 and 500 number of neurons. The power spectra are calculated from a time histogram with bin size 1 ms and the spike count is averaged over the number of neurons. The spectra are calculated without averaging window, but are smoothed by a Gaussian kernel with $\sigma = 3$

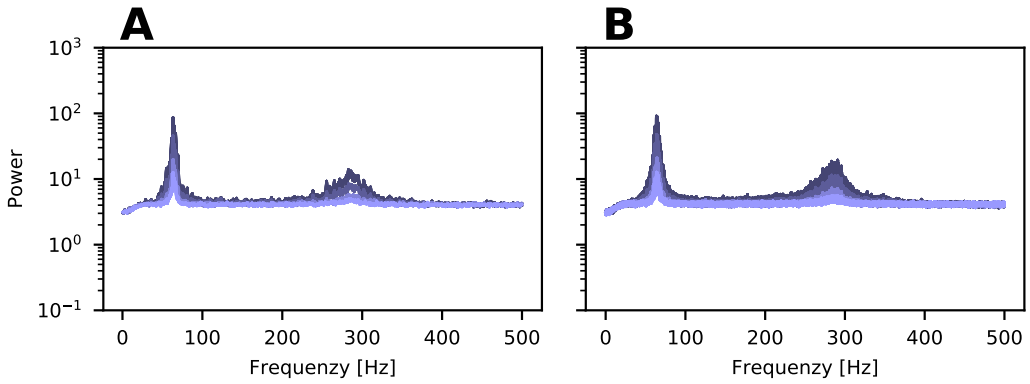


Figure 4.8: Power spectra from subsamples of the spike trains from population L4E of the BDH model. **A**: Power spectrum from 9 seconds simulation time. **B**: Power spectrum from 99 seconds simulation time. Both spectra are calculated without averaging windows and smoothed by a Gaussian kernel with $\sigma = 3ms$.

4.3.1 Subsampling from longer simulation time

In the subsampling from a simulation run for 100 seconds, the same bin size, number of trials and sample sizes were used as in the section above. Only the comparison between the short and long simulation time for population L4E and for 100 trials is shown here, since the differences between the populations and number of trials were similar for the two simulations time. Fig. 4.8 shows the results.

The first thing to note is that the spectrum for the longer simulation time contains a little more noise as seen by the slightly thicker lines. The amplitude of the 300 Hz peak seems slightly larger for the longer simulation as well. Interestingly, the peak at around 80 Hz looks more similar to each other than the 300 Hz peak.

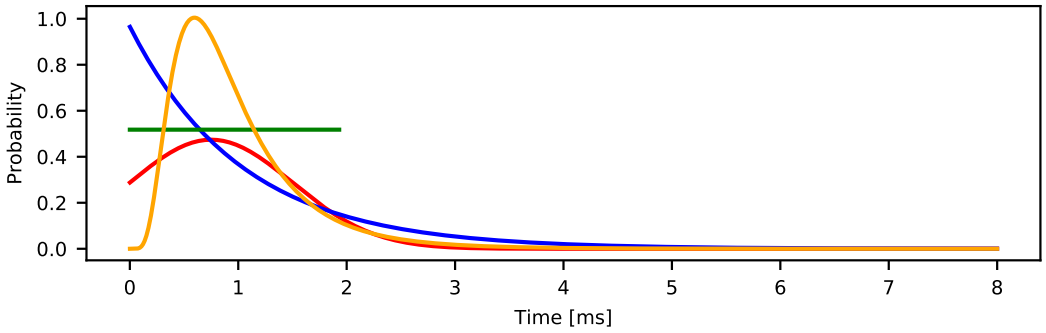


Figure 4.9: Probability density functions for the delay distributions with the parameters in Table 4.1. Red: Truncated gaussian, blue: exponential, green: uniform, orange: lognormal

4.4 Effects of the delay distributions

This section first shows the analytical power spectra for each population in the BDH model for the delay distribution truncated Gaussian, exponential, uniform and lognormal. The parameters for the lognormal distributions were chosen to match the mean and standard deviation of the truncated Gaussian distribution, and the parameters for the exponential and uniform distributions were chosen to match the mean of the truncated Gaussian distribution. The derivations of these parameters are shown in the Appendix and the parameters are listed in Table 4.1. Fig. 4.9 shows the probability density functions for each distribution with these parameters.

Table 4.1: Parameters for the delay distributions of the connections for the inhibitory connections to themselves. The values are chosen to match the mean and or the standard deviation of the truncated Gaussian distribution (see the Appendix)

	Truncated Gaussian	**Exponential**	**Uniform**	**Lognormal**
Parameters	$\mu = 0.75$ ms $\sigma = 0.75$ ms	$\lambda = 1.0356$	$a = 0.0$ ms $b = 1.9314$ ms	$\mu = -0.1959$ ms $\sigma = 0.5673$ ms

With the delay distributions described above, the trajectories of the complex eigenvalues are shown together with the spectra for the dominant term of each eigenmode. Finally, the analytical power spectra were computed for a large number of different mean and standard deviation values for the truncated Gaussian distribution for the delays of the inhibitory connections.

4.4.1 Analytical spectra

Fig. 4.10 shows the analytical power spectra for the eight populations of the BDH model. The red line shows the analytical spectrum using the truncated Gaussian distribution as in the original BDH model. These spectra are in agreement with the spectra shown by Bos et al. [2016].

The high frequency peak is visible in the power spectra for the models using all the other delay distributions except the exponential distribution. This is in agreement with Brunel and Hakim [1999] who states that by using a exponential distribution for the delays, one can get a network dynamic that is always in stable state. An unstable network dynamic would show oscillatory activity. The model with the exponential delay distribution shows a larger amplitude for the 80 Hz peak than the models using the other distributions.

The high frequency peaks for the models using uniform and lognormal distributions are close to the same frequency as for the truncated Gaussian distribution, but the amplitude for the models with lognormal distribution is much larger.

With the parameter used here, the spectra for the model using the uniform distribution are most similar the the ones obtained from the model with the original truncated Gaussian distribution.

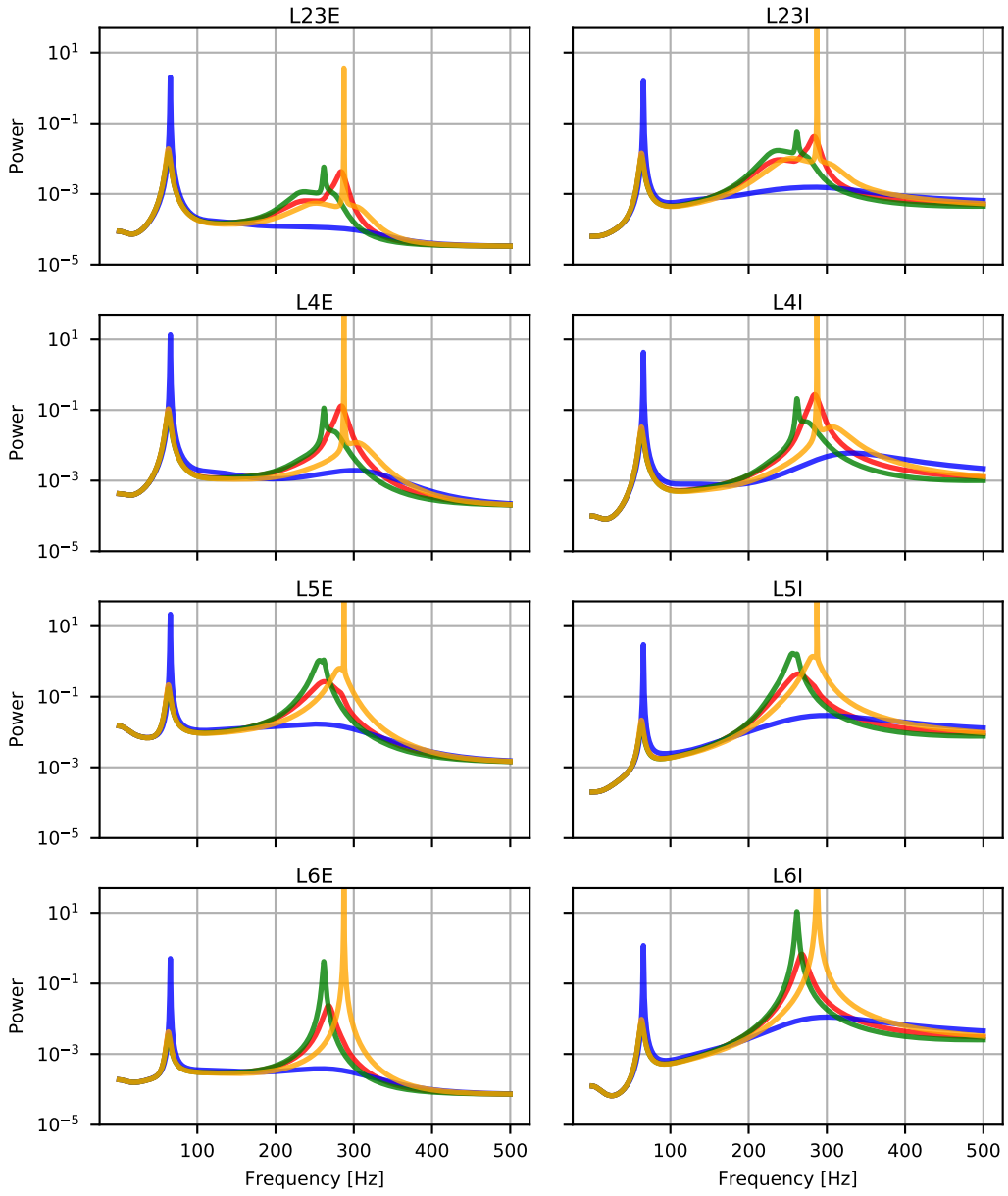


Figure 4.10: Analytical power spectra from all the populations of the BDH model using different delay distributions. Only the delay distributions of the four connections from one inhibitory population to itself are changed, the other delay distributions uses the truncated Gaussian distribution with parameters as in the original model. Red: truncated Gaussian distribution. Blue: exponential distribution. Green: uniform distribution. Orange: lognormal distribution. The parameters are listed in Table 4.1.

4.4.2 Eigenmodes and trajectories of the eigenvalues

The trajectory plots of the 8 eigenvalues in the complex plane for the four models with different delay distributions are shown in Fig. 4.11. The trajectories show the real and imaginary part of the complex eigenvalues of the effective connectivity matrix. Each trajectory is parameterized by ω which is related to the frequency by $\omega = 2\pi f$. The trajectories are plotted for frequencies from 0 Hz shown with black color and and up to 500 Hz shown by lighter colors. Each eigenvalue follows an arc going clockwise and converges towards 0 for high frequencies. The insets show the trajectories near the critical point 1 marked by the blue star.

Fig. 4.12 shows the dominant term $|1/(1-\lambda_i(\omega)|$ for each eigenvalue λ_i as a function of frequency. The figure shows the contribution from each eigenmode to the compound power spectrum.

The dominant term of the eigenmodes $|1/(1-\lambda_i(\omega)|$ is plotted against the frequency and shown in Fig. 4.12. This plots gives the same information as the trajectory plots but it is easier to se how much each eigenmode contribute to the compound spectrum. The analytical power spectra are a combination of the contributions of each of the eigenmodes shown in the figure. Note that the colors of the eigenmodes in Fig. 4.12 corresponds to each of the eight eigenvalues. For each value of omega, the eigenvalues were calculated with the function*numpy.linalg.eigvals*. The output from this function did not necessarily return the eigenvalues in the same order. This is the reason why the color of the eigenmodes sometimes is changed, most notably on the 80 Hz peak from the model using the exponential delay distribution. The colors were kept as they are since it makes it easier distinguish the different eigenmodes and the color changing is minimal.

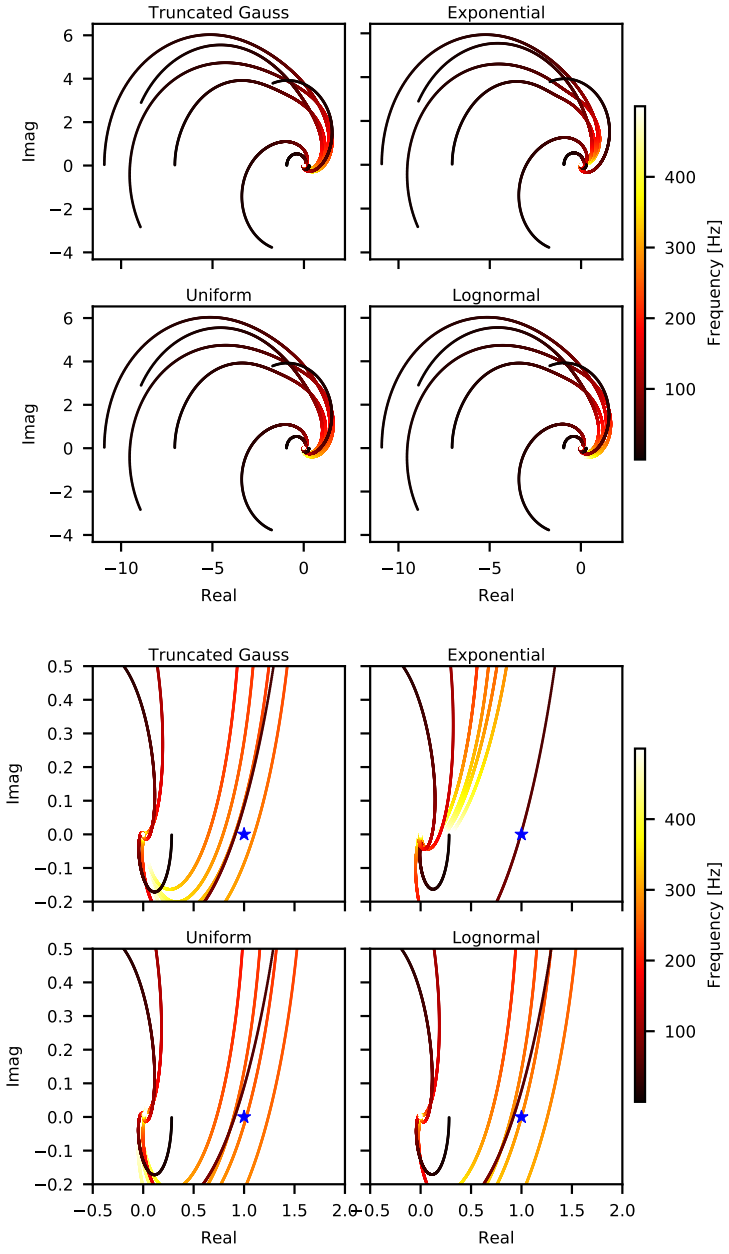


Figure 4.11: Trajectory of the complex eigenvalues of the effective connectivity matrix for the four models with different delay distribution. The real and imaginary values are plotted as a parameterization of the frequency. The colors from dark red towards yellow indicated the frequency of the eigenvalue from 0 Hz to 500 Hz. The blue star is the point 1 in the complex plane.

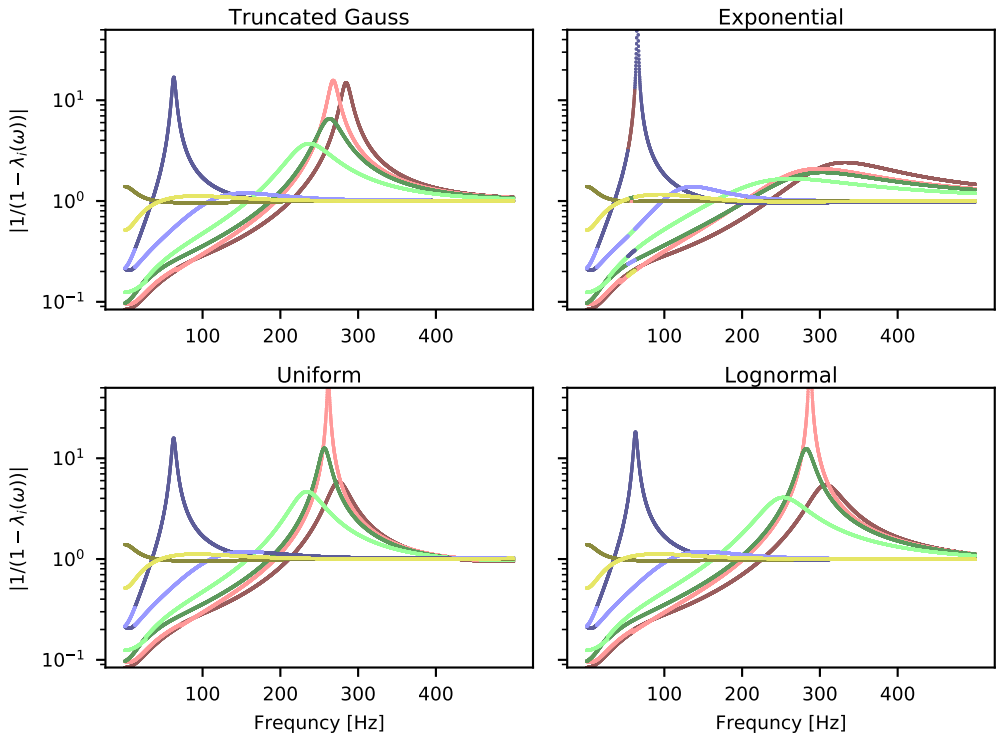


Figure 4.12: The term $|1/(1-\lambda_i(\omega)|$ plotted against frequency for each of the eigenvalues λ_i of the effective connectivity matrix. The parameters of the distributions are listed in Table 4.1.

As seen in Fig. 4.12, the low frequency peak is mostly dominated by only one of the eigenmodes for all the delay distributions, while the the peak at around 300 Hz is a combination of four different eigenmodes. The eigenmode responsible for the 80 Hz peak has a larger amplitude for the model using the exponential delay distribution than the others. This is also seen in the trajectory plots in Fig. 4.11. One of the trajectories for the model using the exponential delay distribution passes very close to the point at around 80 Hz, explaining the higher amplitude for the 80 Hz peak in the power spectrum.

Similarly, one eigenmode passes very close to the critical point for the model using the lognormal distribution, but at a higher frequency. This eigenmode is easily spotted in the Fig. 4.12 as well. Comparing these results with the compound spectra in Fig. 4.10, we see that the increased amplitude for the high frequency peak for the model using the lognormal distribution is largely explained by only one of the eigenmodes.

Looking at the inset of the trajectories in Fig. 4.11, we see that the trajectories for the models using the uniform and lognormal distribution looks most similar. The trajectory for the model using the exponential distribution is most unique since only

one trajectory passes close to the critical point.

4.4.3 Parameter sweep

This section presents the results from the parameter using different mean μ and standard deviation σ of the truncated Gaussian distribution for *all* the inhibitory connections. The two methods used to find the peaks in the power spectrum are described in section 3.6. Since the goal is to see the effect on the dynamics of different delay parameters, the results of only one population is shown. Population L4I was chosen due to the prominent peak in the power spectrum for this population.

Fig. 4.13 **A** shows the analytical power spectra for five parameter combinations, chosen as a mix of high and low mean and standard deviation values. This subfigure shows that the combination of $\mu = 3.9$ ms and $\sigma = 2.5$ ms results in a spectrum with only one peak at around 80 Hz, located by both methods. The use of $\mu = 3.9$ ms and $\sigma = 1.5$ ms is similar but shows a slight peak around 290 Hz in addition which is found by the high-frequency method. The spectrum from using $\mu = 3.9$ ms and $\sigma = 0.2$ ms still shows the low frequency peak at 80 Hz but a high frequency peak at 310 Hz is now present. When the mean and standard deviation both have small values such as $\mu = 0.5$ ms and $\sigma = 0.2$ ms, the low frequency peak is located closer to 60 Hz and the high frequency has larger amplitude and has a frequency over 400 Hz. The final parameter combination showed in the figure is the spectrum obtained from using $\mu = 0.5$ ms and $\sigma = 3.0$. In this spectrum, two peaks are observed at lower frequency. One at around 80 Hz and one at around 120 Hz, where the 120 Hz peak has greater amplitude.

Fig. 4.13 **B** shows the amplitudes of the peaks in the spectra found using the high frequency method. The patterns of light colors corresponds to peaks with large amplitudes, suggesting that certain combinations of the mean and standard deviations produces large peaks. Fig. 4.13 **D** shows the amplitude of the peak found with the low frequency method. The same kind of pattern is observed in this plot, suggesting that some combinations of the mean and standard deviations also produces large peaks at lower frequencies. It looks like some parts of the lighter areas are the same in the two heat maps for values of the mean around 2.6 ms and standard deviation 2.2–3.0 ms, suggesting a possible overlap of the detected peaks for these values.

Fig. 4.13 **C** shows that the peak found by the high frequency method is found for a wide range of frequencies. Certain combinations of μ and σ results in peaks with large frequencies as shown by the whiter areas. The same area is visible in Fig. 4.13 **B** as a black region, indicating a low amplitude of the peaks. The frequency for the low

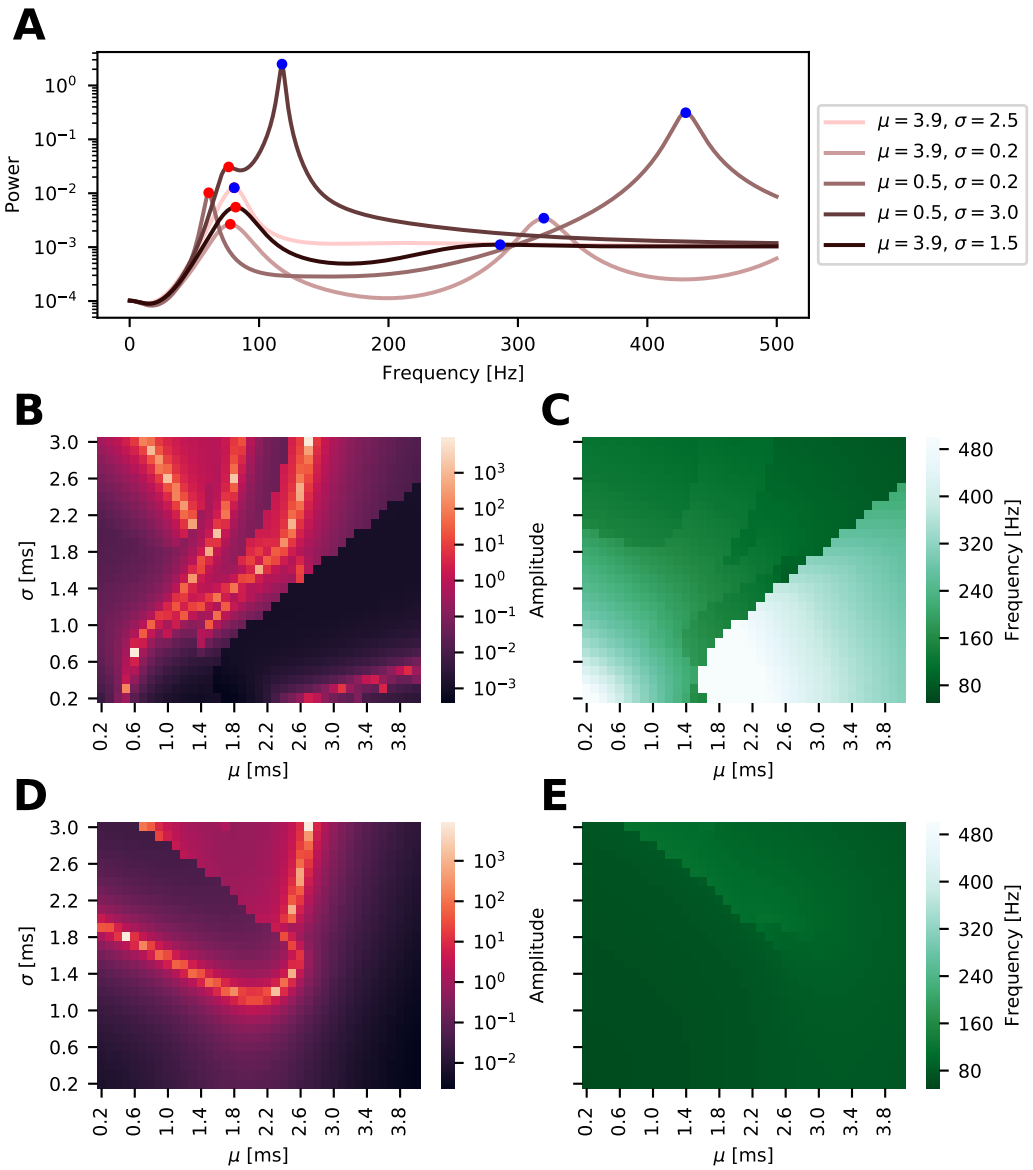


Figure 4.13: Results of the parameter sweep for different mean μ and standard deviation σ parameters for the truncated Gaussian distribution for the delays. The parameters for all the inhibitory connections has been altered (32 in total). Only population L4I is shown. **A**: power spectra for five of the parameter combinations. **B–C**: amplitude (left) and frequency (right) of the high-frequency peak. **D–E**: amplitude (left) and frequency (right) of the low frequency peak.

frequency peak is also changed for some parameter combinations as can be seen in Fig. 4.13 **E**.

4.5 Multi-area model

In this section the results from the multi-are model are presented. The model is compared with the original BDH model and the BDH model using exponential distribution for the delays for the four connections from the inhibitory populations to themselves.

The power spectra for the eight populations of the V1 area of the multi-area model using $\chi = 1$ and $\chi_I = 1$ are shown in Fig. 4.14. None of the power spectra show any clear peaks, suggesting that the oscillatory dynamics are not present.

Fig. 4.15 shows the comparison of the models. Since the multi-area model contains more neurons than the microcircuit model, 21,915 neurons were randomly drawn from the population for all network models and shown in raster plots for comparison. The power spectra from the BDH models are calculated from 9 seconds of simulation time while the multi-area models are calculated from 19 seconds of simulation time. The green line in the power spectrum plots for the BDH model is the analytical power spectra.

The figure shows that both the thin stripes in the raster plot and the high frequency peak in the power spectrum are gone from the BDH model when the exponential distribution is included in the four inhibitory connections. Interestingly, the amplitude of the 80 Hz peak in in the model with exponential delays is lower than the analytical results.

The spectra for the multi-area models shows only a slight bump at around 320 Hz. There is a small peak at around 100 Hz, but the stripes in the raster plots are harder to discern. The peaks in the power spectrum for the multi-area model with $\chi = 1.9$ and $\chi_I = 2.0$ is a little bit more pronounced than the model with $\chi = \chi_I = 1$.

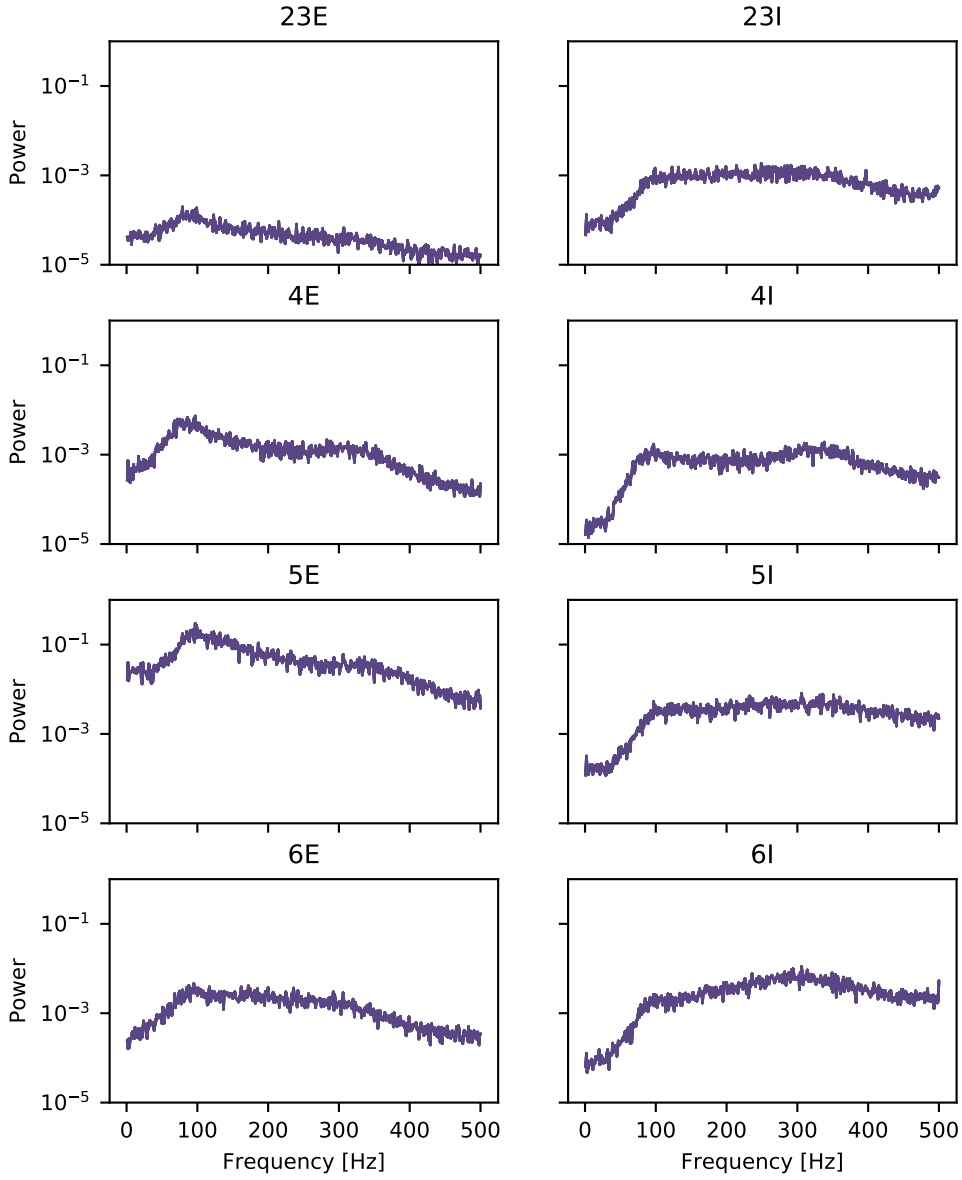


Figure 4.14: Power spectra for the 8 populations from the V1 area in the multi-area model with $\chi = \chi_I = 1$. The spectra are calculated using a time histogram with bin size 1 ms and averaged over time windows of 1000 ms.

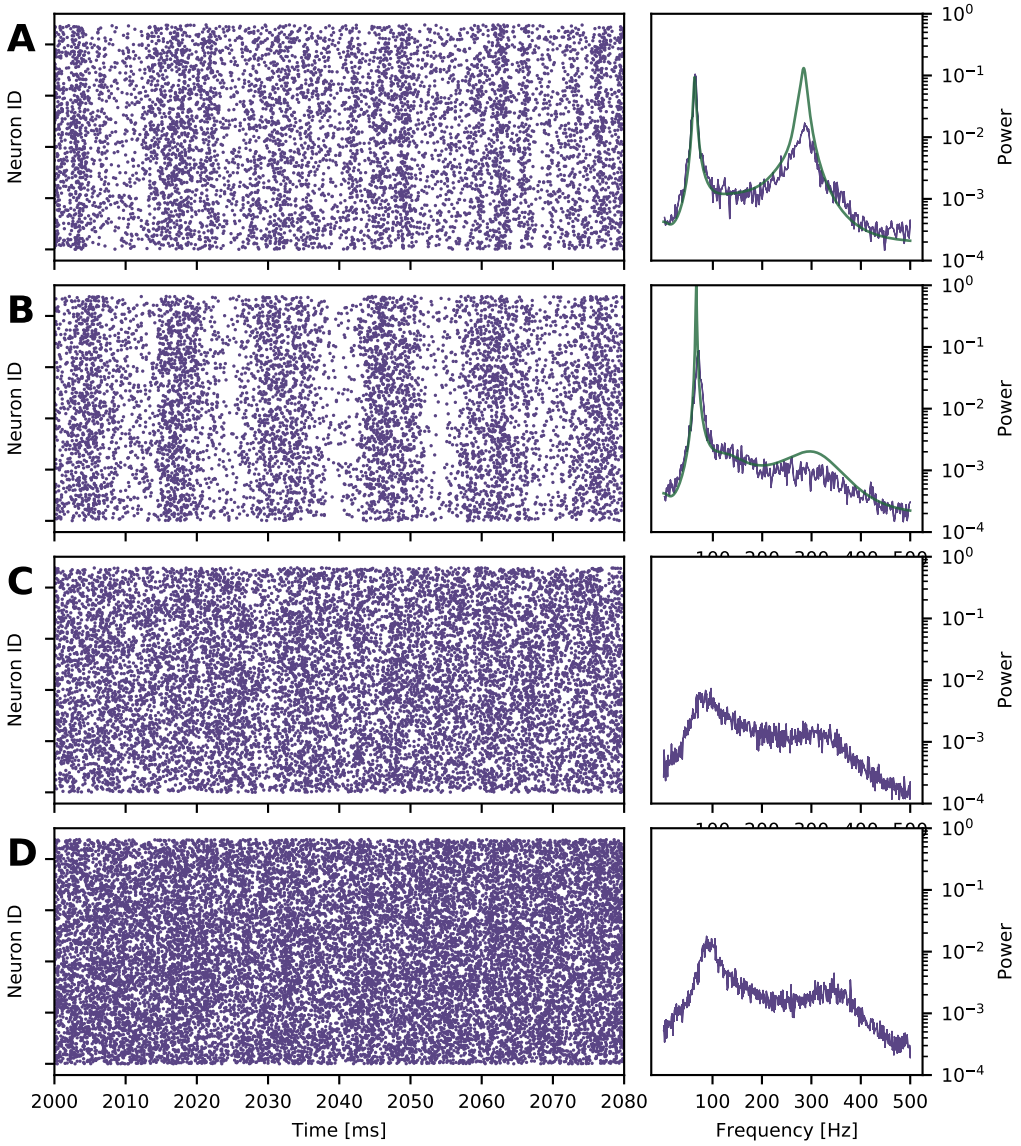


Figure 4.15: Raster plot and power spectra from different models. **A**: population L4E from the original BDH model. **B**: population L4E from the BDH model with exponential delay distribution for the four connections from the inhibitory populations to themselves. **C**: population L4E from the V1 of the multi-area model with $\chi = \chi_{\mathrm{I}} = 1$. **D**: population L4E from the V1 area of the multi-area model with $\chi = 1.9$ and $\chi_{\mathrm{I}} = 2.0$. In the raster-plots, 21,915 neurons have been randomly drawn for each model. The power spectrum is calculated using all the neurons from the population for duration 1–10 seconds for the BDH models and 1–20 seconds for the multi-area models.

Chapter 5

Discussion

In my thesis I have looked at modified versions of the microcircuit model and used different methods to detect the oscillatory activity in order to find out more about the origins the high frequency oscillations observed in spiking network models.

5.1 Different neuron and synapse models

In my analysis, it was showed that the oscillations in the network dynamics are not caused by artifacts in the models due to the simplifications of using discretized time steps or discretized delay values. The time discretization can cause an artificial synchrony in the network dynamics, but it is not the cause of the high frequency oscillations. The use of precise-spike-times neurons and continuous delay synapses does change the dynamics of the network slightly, as seen by the small shift to the right of the high frequency peak in the power spectra in Fig. 4.4. The shift however, is too small to explain the origin of the high frequency oscillations, and the vertical stripes are still visible in the raster plots for all the models.

The use of only continuous delays *without* precise-spike-times neurons also causes a small shift of the high frequency peak in the power spectra, but the shift is to the left. This effect can be explained by how the NEST simulator handles the rounding of delay values in the synapses. When using the standard synapse model, if the delay values assigned to the synapse falls outside the simulation time grid, they will be rounded to the nearest grid point since the synapse model only accept integer multiples of the time resolution as delay values. Assigning the same values to the continuous-delay synapse will result in no rounding since it handels continuous values. However, when using continuous-synapses together with a neuron model that doesn't handle spikes in con-

tinuous time, and the delay values are not integer-multiples of the time resolution, the fraction is ignored, resulting in a rounding *down* of the delay values. When the delays are always rounded down, the result is a distribution with slightly smaller delay values on average. The correct way to use the continuous delay synapses is in combination of the precise-spike-times neurons to avoid this erroneous rounding.

The raster plot for the model using precise-spike-times (green) looks more even than the others. That is most likely due to the small time window of 50 ms plotted and not the effect of the neuron model. A raster plot over longer duration or from other time segments would most likely look more similar to the raster plots from the other models.

5.2 Subsampling

The results from the subsampling of the spike trains show that the peaks in the power spectra can be observed for subsamples of few spike trains. For population L4I the 300 Hz peak becomes visible for as few as 50 neurons when the spectrum is averaged over 10 or 100 trials.

The observation that the peaks become more visible for larger subsamples is in agreement to the arguments about the contributions from the auto-correlations of individual neurons and the cross-correlations between all pair of neurons as explained in section 3.2.2.

One should be a bit careful with the conclusions about how many neurons that needs to be sampled in order to observe the peaks in the power spectra. The peaks are easiest to discern in population L4E, L4I, L5E, L5I and L6I. However the maximum number of neurons in the subsampling was 500, which is about half of the total number of neurons in population L5I. It makes sense to believe that the peak should be visible when as much as 50 % of the population is included in the sample. The largest population is L4E with over 20,000 neurons, but we can still se the peak for a subsample of 500 neurons, which suggests that peaks can be seen for smaller subsamples of larger populations as well, given that the population show large enough oscillatory activity.

Another observation from the subsampling is that the visibility of the peaks matches that of the spectra for the full populations. Peaks with low amplitude like the low frequency peak of populations L6E and L6I is smaller also for the subsamples. This proportionality is not surprising, but it means that peaks that might be found in the power spectra for experimental data needs to be sufficiently large in order to detect them from fewer recorded neurons.

The subsampling from the simulation of 100 seconds yields similar results as the shorter simulations as showed in Fig. 4.8. The high frequency peak for the longer simulation time seems to have slightly larger amplitude than the shorter simulation, but the low frequency peak have the same amplitude. Given that the network dynamics is the same regardless of the duration of the simulation, one would expect that the estimated power spectrum of the population average firing activity would not change by increasing the sampling time. The differences in the height of the high frequency peak is not very large, and is most likely caused by the fact that some neurons are more correlated than others. In the subsampling from over 20,000, there are a lot of possible combinations of 500 neurons. In a subsample with more correlated neurons, the peaks in the power spectrum would be larger. It is interesting to note that the high frequency peak seems to be more sensitive to the selection of neurons than the low frequency peak, but this claim needs to be tested more before any conclusions can be made.

The spectrum for the longer simulation time is also a little broader, meaning that it contains more noise. This could be due to the fact that for the the longer simulation time, the frequencies are sampled at a finer resolution in the power spectrum. For the short simulation, only 9 seconds was used to calculate the power spectrum. Given that the sampling frequency is 1,000 Hz (since the bin size of the time histogram is 1 ms), the totalt number of points in the power spectrum is 9,000. The maximum frequency possible to sample for this sampling frequency is 500 Hz due to the Nyquist frequency (see section 2.6.2). The total number of points in the power spectrum for the frequencies up to 500 Hz in Fig. 4.8 **A** is therefore 4500. For the long simulation, 99 seconds of simulation time are used to calculated the power spectrum. The sampling frequency is the same, yielding a total of 99,000 points of the power spectra. The number of points for frequencies up to 500 Hz is half of that, equal to 49,500. The more points means that the power spectrum will be more smooth, but it is possible that it also shows some noise that would not be detected with a more coarse frequency resolution.

The end conclusion for the longer simulation time is that the detectability of the peaks in the power spectrum is not increased by sampling spike counts for longer periods.

5.3 Analytical results

The use of different distributions for the synaptic delays has an effect on the analytical power spectra. Most notably the disappearance of the high frequency peak when using

an exponential distribution and the large peak when using a lognormal distribution as seen in Fig. 4.10. The probability density functions for each of the distributions are shown in Fig. 4.9. The exponential distribution has a lot of short delays values compared to the other distributions. The parameters for the distributions were chosen to match the mean and/or the standard deviation of the truncated Gaussian distribution. The mean of the truncated Gaussian distribution is close to 1 ms and the standard deviation around 0.63 ms. For the exponential distribution, both the mean and standard deviation is $\frac{1}{\lambda}$, which is close to 1 to when the mean is matched with the truncated Gaussian distribution. This means that the standard deviation is for the exponential distribution is *larger* than the standard deviation for the truncated Gaussian distribution, which could also play a part in the difference of the high frequency peak.

The standard deviation of the uniform distribution is also not possible to match to the truncated Gaussian distribution at the same time as the mean. In this case, the standard deviation of the uniform distribution with $a = 0$ and $b \approx 2$ is about 0.58 ms, which is close to that of the truncated Gaussian distribution. The difference between the models using the truncated Gaussian distribution and the uniform distribution is also not large.

The model using the lognormal distribution shows a large peak in the power spectrum for the high frequency. This is explained by the contribution of one of the eigenmodes due to the fact that one of the eigenvalues is very close to the critical value 1. With a small tuning of the parameters, it might be possible to avoid this singularity of the eigenmode. Since four eigenmodes contribute to the high frequency peak, it is likely that the high frequency peak would still be present even after a small tuning of the parameters.

The method used for these analytical result was presented by Bos et al. [2016]. Here it is showed that the low frequency peak in the power spectra is accurately predicted. However, the high frequency peak is not predicted as accurately. This effect is also seen in Fig. 4.15 for the BDH models. The high frequency peaks for both the standard BDH model and the model with exponential delays has a higher amplitude in the analytical spectra than in the spectra obtained from simulations. The reason given by Bos et al. [2016] is that on the time scale of the lower frequencies, it can be shown that the network behaves like that of an AI state, while on the time scale of the higher frequency, it approaches the SI state. In linear response theory, the network state is assumed to be asynchronous Grytskyy et al. [2013]. For higher frequencies, the dynamics is too synchronous for the theory to be accurate. However, the prediction can still be used to get an estimate of the power spectra for higher frequencies.

The conclusion from this part of the analysis is that only the model with exponential distribution for the delays shows no high frequency peak when the parameters chosen are matched to the ones from the truncated Gaussian distribution. This however, does not mean that it is impossible to suppress the high frequency peak by using other the other distributions with other parameters.

In addition to using different delay distribution, one can change the parameters for the truncated Gaussian distribution to change the analytical power spectrum. The results obtained from a parameter scan of different mean and standard deviations reveals that the dynamics are sensitive to changes to the delay distribution. This is in agreement with what was observed in Bos et al. [2016]. By looking at the heat maps for the high frequency method (Fig. 4.13 **B–C**), the black area in the amplitude plot suggest peaks with low amplitude for these parameters. The border where the black area changes to brighter colors corresponds to the area where the white area changes to the green in the frequency plot. The parameters for the dark areas results in peaks at high frequencies around 450 Hz, while the lighter areas results in peaks for low frequencies around 100 Hz. This change in the frequency position at these parameter is an interesting topic for further study.

An interesting result which is worth looking more into is the bright curves shown in the heat maps for the amplitudes. This suggest that for certain parameters the amplitude gets much larger for both the high and low frequency peaks. The reason is most likely that at least one of the eigenvalues of the effective connectivity map for these parameters is close to the critical point. The question is how the parameters relates to the trajectory of the eigenvalues.

The method of showing the results from the analysis from the parameter sweep has its downsides. The high-frequency method is based on a observation without theoretical support. The way it is implemented will always either find a peak or the boundary points at 0 Hz or 500 Hz, but it does not say anything about whether more peaks are observed in the spectrum or not. There is some parameter combinations that results in the two methods finding the same peak, but that does not necessarily mean that no other peaks are present in the spectrum. From the plots shown in Fig. 4.13 it is not possible to tell.

While the analysis showed here doesn't explain the origin of the observed high frequency oscillations, it shows that the oscillatory activity can be avoided by carefully selecting the delay distributions or delay parameters. The synaptic strengths have also been shown to affect the oscillations in the network as well. Hagen et al. [2016] used a lognormal distribution for the synaptic weights which they found to reduce the

oscillations.

5.4 Multi-area-model

The multi-area model with χ and χ_I equal to 1 contains no high frequency peaks except a small bump in population L4E as seen in Fig. 4.15. The amplitude of the peak at around 80 Hz is somewhat present, but one order of magnitude smaller than the same peak in the power spectrum from the BDH models. Both the BDH models and the area of the multi-area model showed here represent a patch of the visual cortex area V1 so the results can be compared.

The multi-area model with $\chi = 1.9$ and $\chi_I = 2.0$ shows similar power spectrum as the model with $\chi = \chi_I = 1$, but with the peaks slightly more prominent. The change in the χ parameters do affect the high frequency oscillatory activity a little bit, but it seems like it is not very sensitive to changes in these parameters.

The absent of the high frequency peak in the multi-area models suggests that the inclusion of connections between areas reduces the high frequency oscillations. One possible reason why this happens is that the network becomes more heterogeneous and therefor not containing the synchronous activity that generates the oscillations.

Appendix

Fitting delay distribution parameters

The truncation of the Gaussian distribution makes the mean and standard deviation not equal to the parameters used. The result of the truncation is a shift of the mean and standard deviation values. The formulas are described by Burkardt [2014].

For a Gaussian distribution with mean μ_G and standard deviation σ_G truncated at points a and b where $a < b$, the following variables are defined

$$\alpha = \frac{a - \mu_G}{\sigma_G} \tag{5.1}$$

$$\beta = \frac{b - \mu_G}{\sigma_G} \tag{5.2}$$

The mean for the truncated Gaussian distribution μ_t is then defined by

$$\mu_t = \mu_G - \sigma_G \cdot \frac{\phi(\beta) - \phi(\alpha)}{\Phi(\beta) - \Phi(\alpha)}, \tag{5.3}$$

where

$$\phi(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}} \tag{5.4}$$

and $\Phi(x)$ is defined by Eq. (3.25). The variation of the truncated Gaussian distribution σ_t^2 is given by

$$\sigma_t^2 = \sigma_G^2 \cdot \left[1 - \frac{\beta\phi(\beta) - \alpha\phi(\alpha)}{\Phi(\beta) - \Phi(\alpha)} - \left(\frac{\phi(\beta) - \phi(\alpha)}{\Phi(\beta) - \Phi(\alpha)} \right)^2 \right]. \tag{5.5}$$

For the parameters $\mu_G = 0.75$ ms and $\sigma_G = 0.75$ ms used for the truncated Gaussian distribution for the inhibitory delays, the mean and standard deviation becomes $\mu_t \approx 1.0300$ ms and $\sigma_t \approx 0.6348$ ms.

Exponential distrubution

The mean and standard deviation of the exponential distribution are both defined as

$$\frac{1}{\lambda} \tag{5.6}$$

(see Robert V. Hogg [2014]). It is not possible to fit both the mean and standard deviation for this distribution to the truncated Gaussian distribution, therefore only the mean is fitted. For the mean of the truncated Gaussian distribution μ_t and the mean of the exponential distribution μ_e to be equal, lambda is simply

$$\mu_e = \frac{1}{\lambda} = \mu_t \tag{5.7}$$

$$\Rightarrow \lambda = \frac{1}{\mu_t} \tag{5.8}$$

Uniform distribution

The mean of the uniform distribution from points a to b is defined by

$$\mu_u = \frac{a+b}{2}, \tag{5.9}$$

while the standard deviation is defined by

$$\sigma_u = \frac{b-a}{\sqrt{12}}. \tag{5.10}$$

The formulas are taken from Robert V. Hogg [2014]. It is not possible to fit both the means and standard deviation to the truncated Gaussian distribution, so only the mean is fitted. The derivation is as follows

$$\mu_u = \frac{a+b}{2} = \mu_t \Rightarrow a+b = 2\mu_t. \tag{5.11}$$

The lower limit a for the distribution is chosen to be 0 to allow as small values of the delays as the truncated Gaussian distribution. That leaves the parameter b for the high limit to be determined as $b = 2\mu_t$.

Lognormal distribution

The mean of the lognormal distribution is defined by

$$\tilde{\mu}_{\log} = e^{\mu_{\log} + \frac{\sigma_{\log}^2}{2}}, \tag{5.12}$$

where $\mu_{\log}$ and $\sigma_{\log}$ are the parameters μ and σ for the lognormal distribution and $\tilde{\mu}_{\log}$ is the mean value [Press et al., 2007, ch. 6.14 p.320–339].

The standard deviation is defined by

$$\tilde{\sigma}_{\log}^2 = \left(e^{\sigma_{\log}^2} - 1\right) e^{2\mu_{\log} + \sigma_{\log}^2}, \tag{5.13}$$

where $\tilde{\sigma}_{\log}$ is the standard deviation and $\mu_{\log}$ and $\sigma_{\log}$ again is the parameters for the lognormal distribution.

The two parameters can be fitted so that the mean and standard deviation of the log-normal distribution is equal to that for the truncated Gaussian distribution by solving the two equations with respect to $\mu_{\log}$ and $\sigma_{\log}^2$

$$e^{\mu_{\log} + \frac{\sigma_{\log}^2}{2}} = \mu_{\mathrm{t}} \tag{5.14}$$

$$\left(e^{\sigma_{\log}^2} - 1\right) e^{2\mu_{\log} + \sigma_{\log}^2} = \sigma_{\mathrm{t}}. \tag{5.15}$$

Eq. (5.14) can be written as

$$\sigma_{\log}^2 = 2(\ln \mu_{\mathrm{t}} - \mu_{\log}). \tag{5.16}$$

Inserting this for $\sigma_{\log}^2$ in Eq. (5.15) yields

$$\mu_{\log} = 2\ln \mu_{\mathrm{t}} - \frac{\ln\left(\sigma_{\mathrm{t}}^2 + e^{2\ln \mu_{\mathrm{t}}}\right)}{2} \tag{5.17}$$

This equation can in turn be inserted into Eq. (5.16) to get an expression for $\sigma_{\log}^2$

$$\sigma_{\log}^2 = \ln\left(\sigma_{\mathrm{t}}^2 + e^{2\ln \mu_{\mathrm{t}}}\right) - 2\ln \mu_{\mathrm{t}} \tag{5.18}$$

Bibliography

Frederico A.C. Azevedo, Ludmila R.B. Carvalho, Lea T. Grinberg, José Marcelo Farfel, Renata E.L. Ferretti, Renata E.P. Leite, Wilson Jacob Filho, Roberto Lent, and Suzana Herculano-Houzel. Equal numbers of neuronal and nonneuronal cells make the human brain an isometrically scaled-up primate brain. *Journal of Comparative Neurology*, 513(5):532–541, 2009. doi: 10.1002/cne.21974.

Hannah Bos, Markus Diesmann, and Moritz Helias. Identifying Anatomical Origins of Coexisting Oscillations in the Cortical Microcircuit. *PLoS Computational Biology*, 12(10):1–34, 2016. doi: 10.1371/journal.pcbi.1005132.

Thomas Brochier, Lyuba Zehl, Yaoyao Hao, Margaux Duret, Julia Sprenger, Michael Denker, Sonja Grün, and Alexa Riehle. Data Descriptor: Massively parallel recordings in macaque motor cortex during an instructed delayed reach-to-grasp task. *Scientific Data*, 5:1–23, 2018. doi: 10.1038/sdata.2018.55.

Nicolas Brunel. Dynamics of sparsely connected networks of excitatory and inhibitory neurons. *Computational Neuroscience*, 8(5-6):183–208, 2000. doi: 10.1016/S0928-4257(00)01084-6.

Nicolas Brunel and Vincent Hakim. Fast Global Oscillations in Networks of Integrate-and-Fire Neurons with Low Firing Rates. *Neural Computation*, 1671:1621–1671, 1999.

John Burkardt, October 2014. URL https://people.sc.fsu.edu/~jburkardt/presentations/truncated_normal.pdf. [Last accessed[2019-05-13].

György Buzsáki and Xiao-Jing Wang. Mechanisms of Gamma Oscillations. *Annual Review of Neuroscience*, 35(1):203–225, 2012. doi: 10.1146/annurev-neuro-062111-150444.

DLMF. *NIST Digital Library of Mathematical Functions*. http://dlmf.nist.gov/, Release 1.0.22 of 2019-03-15, 2019. URL http://dlmf.nist.gov/. F. W. J. Olver, A. B. Olde Daalhuis, D. W. Lozier, B. I. Schneider, R. F. Boisvert, C. W. Clark, B. R. Miller and B. V. Saunders, eds.

Elephant. Elephant toolkit, 2019. URL https://elephant.readthedocs.io/en/latest/. [Last accessed: 2019-05-10].

Elephant Kernels. Elephant toolkit, 2019. URL https://elephant.readthedocs.io/en/latest/reference/kernels.html. [Last accessed 2019-05-13].

Nicolas Fourcaud and Nicolas Brunel. Dynamics of the Firing Probability of Noisy Integrate-and-Fire Neurons. *Neural Computation*, 14(9):2057–2110, 2002. doi: 10.1162/089976602320264015.

S. Garcia, D. Guarino, F. Jaillet, T.R. Jennings, R. Pröpper, P.L. Rautenberg, C. Rodgers, A. Sobolev, T. Wachtler, P. Yger, and A.P. Davison. Neo: an object model for handling electrophysiology data in multiple formats. *Frontiers in Neuroinformatics*, 8:10, February 2014. doi: 10.3389/fninf.2014.00010.

Wulfram Gerstner and Werner Kistler. *Spiking Neuron Models*, chapter Statistics of spike trains, pages 150–163. Cambridge University Press, 2002.

Dmytro Grytskyy, Tom Tetzlaff, Markus Diesmann, and Moritz Helias. A unified view on weakly correlated recurrent networks. *Frontiers in Computational Neuroscience*, 7(October):1–19, 2013. doi: 10.3389/fncom.2013.00131.

Espen Hagen, David Dahmen, Maria L. Stavrinou, Henrik Lindén, Tom Tetzlaff, Sacha J. Van Albada, Sonja Grün, Markus Diesmann, and Gaute T. Einevoll. Hybrid scheme for modeling local field potentials from point-neuron networks. *Cerebral Cortex*, 26(12):4461–4496, 2016. doi: 10.1093/cercor/bhw237.

Alexander Hanuschkin, Susanne Kunkel, Moritz Helias, Abigail Morrison, and Markus Diesmann. A General and Efficient Method for Incorporating Precise Spike Times in Globally Time-Driven Simulations. *Frontiers in Neuroinformatics*, 4(October):1–19, 2010. ISSN 1662-5196. doi: 10.3389/fninf.2010.00113.

M. R. Jarvis and P. P. Mitra. Sampling Properties of the Spectrum and Coherency of Sequences of Action Potentials. *Neural Computation*, 13(4):717–749, 2001. doi: 10.1162/089976601300014312.

Fredrik Johansson et al. *mpmath: a Python library for arbitrary-precision floating-point arithmetic (version 0.18)*, December 2013. `http://mpmath.org/`.

Eric Jones, Travis Oliphant, Pearu Peterson, et al. SciPy: Open source scientific tools for Python, 2001–. URL `http://www.scipy.org/`. [Last accessed: 2019-05-13].

Erwin Kreyszig. *Advanced Engineering Mathematics*, chapter 11.9, pages 522–533. John Wiley and sons, inc, 2011.

Jeyashree Krishnan, PierGianLuca Porta Mana, Moritz Helias, Markus Diesmann, and Edoardo Di Napoli. Perfect spike detection via time reversal. *Frontiers in Neuroinformatics*, 11(January), 2017. doi: 10.3389/fninf.2017.00075.

Roberto Lent, Frederico A.C. Azevedo, Carlos H. Andrade-Moraes, and Ana V.O. Pinto. How many neurons do you have? Some dogmas of quantitative neuroscience under revision. *European Journal of Neuroscience*, 35(1):1–9, 2012. doi: 10.1111/j.1460-9568.2011.07923.x.

Abigail Morrison, Sirko Straube, Hans Ekkehard Plesser, and Markus Diesmann. On graminicolous pyrenomycetes from Fennoscandia. 1. Dictyosporous species, 2007.

NEST. Nest features, 2019. URL `https://www.nest-simulator.org/features/`. [Last accessed: 2019-05-10].

Travis E. Oliphant. NumPy: A guide to numpy, 2006–. URL `http://www.numpy.org/`. [Last accessed: 2019-05-13].

Alexander Peyser, Ankur Sinha, Stine Brekke Vennemo, Tammo Ippen, Jakob Jordan, Steffen Graber, Abigail Morrison, Guido Trensch, Tanguy Fardet, Håkon Mørk, Jan Hahne, Jannis Schuecker, Maximilian Schmidt, Susanne Kunkel, David Dahmen, Jochen Martin Eppler, Sandra Diaz, Dennis Terhorst, Rajalekshmi Deepu, Philipp Weidel, Itaru Kitayama, Sepehr Mahmoudian, David Kappel, Martin Schulze, Shailesh Appukuttan, Till Schumann, Hünkar Can Tunç, Jessica Mitchell, Michael Hoff, Eric Müller, Milena Menezes Carvalho, Barna Zajzon, and Hans Ekkehard Plesser. Nest 2.14.0, October 2017. URL `https://doi.org/10.5281/zenodo.882971`.

Tobias C. Potjans and Markus Diesmann. The cell-type specific cortical microcircuit: Relating structure and activity in a full-scale spiking network model. *Cerebral Cortex*, 24(3):785–806, 2014. doi: 10.1093/cercor/bhs358.

William H. Press, Saul A. Teukolsky, William T. Vetterling, and Brian P. Flannery. *Numerical Recipes*. Cambridge University Press, 3. edition, 2007.

Allen T. Craig Robert V. Hogg, Joeseph McKean. *Introduction to Mathematical Statistics*. Pearson Education Limited, 7. edition, 2014.

Maximilian Schmidt. *Modeling and simulation of multi-scale spiking neuronal networks*. PhD thesis, Aachen University, 2016.

Maximilian Schmidt, Rembrandt Bakker, Claus C. Hilgetag, Markus Diesmann, and Sacha J. van Albada. Multi-scale account of the network structure of macaque visual cortex. *Brain Structure and Function*, 223(3):1409–1435, 2018a. doi: 10.1007/s00429-017-1554-4.

Maximilian Schmidt, Rembrandt Bakker, Kelly Shen, Gleb Bezgin, Markus Diesmann, and Sacha Jennifer van Albada. A multi-scale layer-resolved spiking network model of resting-state dynamics in macaque visual cortical areas. *PLoS Computational Biology*, 14(10):1–38, 2018b. doi: 10.1371/journal.pcbi.1006359.

Jannis Schuecker, Markus Diesmann, and Moritz Helias. Modulated escape from a metastable state driven by colored noise. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics*, 92(5):1–11, 2015. doi: 10.1103/PhysRevE.92.052119.

Jannis Schuecker, Maximilian Schmidt, Sacha J. van Albada, Markus Diesmann, and Moritz Helias. Fundamental Activity Constraints Lead to Specific Interpretations of the Connectome. *PLoS Computational Biology*, 13(2):1–25, 2017. doi: 10.1371/journal.pcbi.1005179.

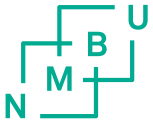
Julius O. Smith. *Mathematics of the Discrete Fourier Transform (DFT) with Audio applications*. BookSurge Publishing, 2. edition, May 2008.

David Sterratt, Bruce Graham, Andrew Gillies, and David Willshaw. *Principles of Computational Modelling in Neuroscience*. Cambridge University Press, 2011. doi: 10.1017/CBO9780511975899.003.

Tom Tetzlaff, Moritz Helias, Gaute T. Einevoll, and Markus Diesmann. Decorrelation of Neural-Network Activity by Inhibitory Feedback. *PLoS Computational Biology*, 8(8), 2012. doi: 10.1371/journal.pcbi.1002596.

Richard F. Thompson. *The Brain: A Neuroscience Primer*. Worth Publishers, 2000.

List of Figures



Norges miljø- og biovitenskapelige universitet
Noregs miljø- og biovitskapelege universitet
Norwegian University of Life Sciences

Postboks 5003
NO-1432 Ås
Norway