Norwegian University
of Life Sciences

**Master's Thesis 2022    30 ECTS**
Faculty of Science and Technology
Professor Hans Ekkehard Plesser

# Exploration of delay discretization and rounding on neuronal network dynamics

Nida Grønbekk
MSc Data Science

# Abstract

It is of great interest to try and simulate the neural activity in the human brain. This way one could possibly create models to help in diagnosing and treating disorders of the brain. This could lead to decreasing the costs of disorders of the brain. In order to make sure we can trust the simulations, several aspects have to be tested and investigated. In this thesis we want to explore how the rounding of delay values in neurons affect the spiking activity of the network as a whole. Is there a more correct way of rounding the delay values? And if we want to make simulations computationally more effective by discretizing both delay values and times, what resolution is necessary in order to achieve spiking activity similar to that of the exact solution?

We try to answer these questions by performing network simulations of both the Brunel and Microcircuit network, for three different models. The first model is a continuous model with continuous delay values and times. This will be viewed as the exact model. In addition we will look into two discrete models with discrete delay values and times. One where each delay value have the same probability of being drawn, and one where the end points of the interval only have half the probability of being drawn. The two discrete models are simulated for several choices of resolution. They will be compared by their statistics, calculated based on the spike times. In addition to comparing the statistics visually, we calculate distance measures based on these statistics.

In the results we see that there is no proof of one of the two rounding rules of delay values being more correct in the sense of being closer to the exact model based on the statistics. That is, whether the minimum and maximum delay value have the same probability of being drawn as the others or not, does not make a difference. We also observe that in order to get activity in the discrete models to be similar to that of the reference model, we need to use a resolution of $\frac{1}{64}$. Using this resolution for the discrete models is computationally less expensive than simulating the continuous model in the case of the Microcircuit network. However this does not apply to the Brunel network.

# Contents

# List of Figures

# List of Tables

# Chapter 1

# Introduction

The human brain is complex. It consists of approximately 86 billion neurons, which are all functioning (Wnuk et al. 2018). Each neuron communicates with thousands of other neurons via electrical signals, creating countless electrical circuits which together with nerves throughout our bodies form our nervous system (Wnuk et al. 2018). The signals controlling our body and functions, and also our thoughts and emotions are sent through the nervous system. With such complex structures and interconnected parts, complications follows. There are endless of ways things can go wrong (Wnuk et al. 2018). These errors lead to disorders of the brain, such as Alzheimer's, depression, addictive disorders, anxiety disorders and so on (Wnuk et al. 2018; Gustavsson et al. 2011).

Disorders of the brain is a problem within health care, not only on the personal level of the patient living with the disorders, but also for society as a whole. In a study performed in 2010 by Gustavsson et al. (2011), it was estimated that the total cost of disorders of the brain in the countries of Europe, including a population of 514 million people, was a total of 798 billion euros. This cost covers 19 major groups of disorders. There were limitations to the data available, so the estimate is considered to be conservative. They estimate that the cost for a single person lies between 285 and 30000 euros, ranging from a headache to neuromuscular disorders. In addition to the financial burden, disorders of the brain also put an emotional and a social burden on the patients and their surroundings (Wnuk et al. 2018; Gustavsson et al. 2011).

In order to someday understand the human brain and its functionalities more thoroughly, and thereby understand things that go wrong in the brain with its interconnected networks, it is of interest to try and model it. If one could accurately simulate and recreate the activity in the brain, it could help to use these models to assist in treating and diagnosing disorders of the brain. This could possibly lead to minimizing the costs of disorders of the brain, which, according to Gustavsson et al. (2011), is the number one economic challenge for European health care, now and in the future.

## 1.1 Goal of the thesis

The developers of the NEST (Deepu et al. 2021) simulator, have created a simulator that tries to recreate spiking activity in the brain. They have focused on recreating the dynamics, size and structure of neuronal networks, rather than recreating exact models of single neurons. In order to trust the simulations, several things have to be explored and tested. One aspect that needs to be explored is the rounding of delay values. By delay values, one refers to the time it takes for a signal emitted from one neuron to arrive the connecting neuron. These values are drawn from a given interval, and in some cases, one wishes to round these values to a selection of grid points. These grid points are chosen such that the minimum and maximum value of the interval is included as grid points. But when we draw these values, the points at each end of the interval only have half the probability of being drawn, compared to the points in between. Should each grid point be given the same probability of being drawn? Or does it not matter? What is more correct?

In addition to the discretization of the delay values, it is of interest to discretize the time as well. By discretizing the time, it means that the updates of neurons in the network, in addition to absorption and emission of spikes, are restricted to the time grid points (Morrison et al. 2007). The temporal spacing between the time grid points is referred to as computation time step $h$. According to Morrison et al. (2007), when one reduces the computation time step $h$, the simulation results converge to the exact solution. However, when the time step is reduced, the simulation time and memory requirements increases. It is therefore of interest to explore what choices of the time step $h$ are small enough to give representative spiking activity, not far from the exact solution, while still being computationally less expensive than when the time is not discretized.

## 1.2 Structure of the thesis

Following this introduction, in Ch. 2, we will introduce the theoretical background needed to understand the results. We introduce the neuron model we will use and then the network models consisting of this neuron. We then introduce and describe several statistics that can be calculated based on the spike times of the neurons in the networks, and then finally a distance measure which is based on these statistics. In Ch. 3, we go into detail about how the delay values are drawn for the different network models. We also explain how the simulations are performed and with what tools. When all the background information is in place, we present the results in Ch. 4. Based on the results, we discuss them and draw a conclusion in Ch. 5.

As mentioned earlier, in order to figure out what way of rounding of the delay values is more correct, and how high the resolution needs to be in order to achieve presentable spiking activity in accordance to the exact solution, two network models will be explored. The Brunel and the Microcircuit network. For each network, simulations will be performed for three different models. One where the delay values and times are continuous, the other two with discrete times and delay values. For the two discrete models, the delay values will be drawn from intervals containing the same grid points. However, for one model the probability of drawing each value is the same, while for the other the end points only have half the probability of being drawn. The two discrete models will be simulated for several choices of resolution. To compare the three models, several statistics will be calculated based on the spike times. The statistics will be compared to see which of the two discrete models are closer to the continuous one, and also to get an idea of which resolution is needed for the two discrete models to achieve similar spiking activity as in the continuous model. This will be done for the two network models separately.

# Chapter 2

# Theory and Background

We will now present the necessary theory needed to perform the simulations and compare them. We start by introducing the neuron as it appears in nature, and the model we use to simulate it.

## 2.1 Neuron

A single neuron consists of three parts. The dendrites, the soma, and the axon. It receives electrical signals from other connecting neurons through the dendrites. The signal then travels through the soma, and if the signal changes the potential inside the soma enough to trigger an action potential, the signal will be sent on down the axon. It then reaches the axon terminals, where it will travel to the connecting neuron via neurotransmitters, and arrive at the receiving neurons dendrites. A neuron can be either excitatory or inhibitory. That is, it can either drive other neurons towards firing, or suppress the activity of connecting neurons (Sterratt et al. 2011).

To make networks of neurons possible to model, one uses a simplified model of neurons. One example is the leaky integrate-and-fire neuron. This model describes the change in the membrane potential without including spatial information, e.g. how the action potential propagates down the axon. According to Sterratt et al. (2011, Ch. 8), the change in membrane potential is described by

$$C_m \frac{dV}{dt} = -\frac{V - E_m}{R_m} + I(t) \ . \tag{2.1}$$

In this model the neuron has the characteristics of a RC-circuit, where $C_m$ describes the capacitance, $E_m$ the equilibrium potential, $R_m$ the resistance, $I(t)$ the input current and $V$ the membrane potential.

The neuron receives input current $I(t)$, and if the potential $V$ of the neuron reaches a threshold value $\theta$, the neuron will fire an action potential. When the spike is fired, the membrane potential of the neuron is set back to the resting potential $E_m$, and the potential is insensitive to new stimulus during a refractory period $\tau_{rp}$. This refractory period prevents the neuron from firing infinitely many spikes. The leak term of equation 2.1, that is, the first term on the right hand side, drives the potential toward the resting potential as time passes if no new input current arrives the neuron.

## 2.2 Brunel Network

We want to use the above mentioned neuron model in simulations of networks of neurons. One of the neural networks that we will take into consideration is the Brunel network. In this network we do not consider spatial placement of neurons. The network consists of leaky integrate-and-fire (LIF) neurons. These will be found in two populations, an excitatory and an inhibitory population. That is, a population consisting of excitatory neurons, and a population consisting of inhibitory neurons. The model tries to represent the real spiking activity in the brain by preserving the same ratio of excitatory and inhibitory neurons in the model as can be found in the brain. That is, we have 80% excitatory, and 20% inhibitory neurons. According to Brunel (2000), each neuron in the model receives $C$ randomly chosen connections from other neurons in the network. From these $C$ connections, a specific number of them come from excitatory neurons, while the rest comes form inhibitory neurons. The number of connections each neuron receives from excitatory and inhibitory neurons is quite small compared to the total number of excitatory and inhibitory neurons present in the network, respectively. It is thereby a sparsely connected network.

We call the current received from the connections to other neurons in the network the recurrent input. This is because the current comes from within the network. As the neurons in the human brain are also affected by spiking in neurons outside the network, but nearby in placement, the model incorporates an external current input to each neuron. For simplicity, it is assumed that the excitatory and inhibitory neurons have the same neuron characteristics, e.g. firing threshold, refractory period, and resting membrane potential. The terms which can be seen in model A in the article by Brunel (2000).

The input current to the neuron $i$ is described through $I_i(t)$. It incorporates input from other neurons in the network, so-called recurrent input, and input from neurons firing in other networks nearby, the external input (Brunel 2000). The input current is modelled by

$$RI_i(t) = \tau \sum_j J_{ij} \sum_k \delta(t - t_j^k - D) \ . \tag{2.2}$$

The variable $R$ represents the resistance to current flowing out of the neuron, and $\tau$ represents the membrane time constant, $\tau = RC$, where $C$ is the membrane capacitance. In the equation, the first sum on the right hand side sums over the synapses, connections between neuron $i$ and neuron $j$, with post-synaptic potential amplitude $J_{ij}$ (Brunel 2000). In our case we use the same postsynaptic potential amplitude $J_{ij} = J > 0$ for synapses from excitatory neurons, both from the ones in the network, and from the excitatory neurons in other networks nearby. The synapses from inhibitory neurons are assigned postsynaptic potential amplitude $J_{ij} = -gJ$.

The second sum in equation 2.2 is over the spikes emitted from connecting neuron $j$. This incorporates both recurrent and external connections. The spikes are emitted at time $t_j^k$ and arrives at neuron $i$ after a time $D$. The time it takes for a spike emitted from neuron $j$ to reach neuron $i$ is referred to as a delay, $D$. The spike times of the external neurons are generated by a Poisson process with rate $\nu_{ext}$.

The synapses are modelled using the delta function $\delta$. The delta function (Kanwal 1983) has the properties

$$\delta(x - \xi) = 0, x \neq \xi \ . \tag{2.3}$$

$$\int_{-\infty}^{\infty} \delta(x - \xi) dx = 1 \ . \tag{2.4}$$

As seen in the article by Brunel (2000), the choice of parameters decides in what kind of state the neuronal spiking would be. The network can be in four different states. Asynchronous irregular (AI), asynchronous regular (AR), synchronous regular (SR) and synchronous irregular (SI). A network is in asynchronous state if the instantaneous firing rate $\nu$ is constant in time. Instantaneous firing rate can be defined by

$$\nu(t) = \frac{1}{N\Delta t} \sum_{i=1}^{N} S_i(t) \ . \tag{2.5}$$

In this equation, $N$ is the number of neurons, and $S_i(t)$ is the number of spikes emitted from neuron $i$ in the time window $[t, t + \Delta t]$ (Sterratt et al. 2011, Ch. 9). If $\nu$ varies in time the network is said to be synchronous. By regular and irregular state one refers to regular or irregular firing of individual neurons in the network. It was shown in the article by Brunel (2000) that there is irregular firing when the parameter $g > 4$, and regular firing when $g < 4$. That is the parameter that determines the postsynaptic potential amplitude of the inhibitory synapses, determining the balance between inhibition and excitation in the network. We get irregular firing when inhibition dominates, and regular firing when excitation dominates. In our simulations we chose parameters to achieve the asynchronous irregular (AI) state. That is, the instantaneous firing rate $\nu$ is constant in time, while individual neurons fire irregularly, which means that inhibition dominates.

A more detailed description of the Brunel network and the parameters which were used in our simulations are given in the following tables A-F. They are very similar to that found in the thesis by Dasbach (2020) and is written in tables after Nordlie et al. (2009).

| A | Model Summary |
|---|---|
| **Populations** | Three: excitatory, inhibitory, external input |
| **Topology** | no spatial information |
| **Connectivity** | fixed number of incoming connections per neuron (fixed in-degree) |
| **Neuron model** | Leaky integrate-and-fire, fixed voltage threshold, fixed absolute refractory time (voltage clamp) |
| **Synapse model** | $\delta$-current inputs (discontinuous voltage jumps) |
| **Input** | Independent fixed-rate Poisson spike trains to all neurons |
| **Measurements** | Spike times |

| B | Populations | |
|---|---|---|
| **Name** | **Elements** | **Size** |
| E | Iaf neuron | $N_{\mathrm{E}} = 4N_{\mathrm{I}}$ |
| I | Iaf neuron | $N_{\mathrm{I}}$ |
| $\mathrm{E}_{\mathrm{ext}}$ | Poisson generator | 1 |

| C | | | Connectivity |
|---|---|---|---|
| **Name** | **Source** | **Target** | **Pattern** |
| EE | E | E | random independent connections, fixed in-degree $C_E$, weight $J_{ex}$, uniformly distributed delay values |
| IE | E | I | random independent connections, fixed in-degree $C_E$, weight $J_{ex}$, uniformly distributed delay values |
| EI | I | E | random independent connections, fixed in-degree $C_I$, weight $J_{in}$, uniformly distributed delay values |
| II | I | I | random independent connections, fixed in-degree $C_I$, weight $J_{in}$, uniformly distributed delay values |
| Ext | $\mathrm{E}_{\mathrm{ext}}$ | E ∪ I | one generator connected to all neurons, weight $J_{ex}$, no delay |

| D | Neuron and Synapse Model |
|---|---|
| **Name** | Iaf neuron |
| **Type** | Leaky integrate-and-fire, $\delta$-current input |
| **Subthreshold dynamics** | $\tau \frac{dV}{dt} = -V(t) + RI(t) \quad \text{if} \quad t \notin [t_i^k, t_i^k + \tau_{\mathrm{rp}})$<br>$V(t) = V_{\mathrm{r}} \quad \text{else}$<br>$V(t=0) = 0$ |
| **Synapse current** | $RI_i(t) = \tau \sum_j J_{ij} \sum_k \delta(t - t_j^k - D)$,<br>constant weight $J_{ij} = J$ for excitatory and $J_{ij} = -gJ$ for inhibitory synapses, delay values drawn from uniform distribution $[d_{\min}, d_{\max}]$ |
| **Spiking** | The neuron spikes at time $t_i^k$ if $V(t_i^k) \geq \theta$. |

| **E** | **Input** |
|---|---|
| **Type** | **Description** |
| Poisson generators | Fixed rate $p_{\text{rate}}$, one generator connected to all neurons |

| **F** | | **Parameters** |
|---|---|---|
| **Name** | **Value** | **Description** |
| $N_E$ | 10000 | number of excitatory neurons |
| $N_I$ | 2500 | number of inhibitory neurons |
| $C_E$ | 1000 | number of incoming excitatory synapses per neuron |
| $C_I$ | 250 | number of incoming inhibitory synapses per neuron |
| $\tau_m$ | 20 ms | membrane time constant |
| $\tau_{ref}$ | 2 ms | absolute refractory period |
| $C_m$ | 1 pF | membrane capacitance |
| $V_r$ | 0 mV | reset potential |
| $\theta$ | 20 mV | spike threshold |
| $V_m$ | 0 mV | initial membrane potential of all neurons |
| $E_L$ | 0 mV | leak reversal potential of neurons |
| $J_{ex}$ | 0.1 | excitatory synaptic strength |
| $J_{in}$ | $-0.5$ | inhibitory synaptic strength |
| $p_{\text{rate}}$ | 20000 spikes/s | rate of poisson generator |
| $T$ | 10 s | Simulation duration |
| $d_{\text{min}}$ | 1.0<br>$1.0 - \frac{h}{2}$ | minimum delay value, droop rounding rule<br>minimum delay value, equal rounding rule, resolution $h$ |
| $d_{\text{max}}$ | 2.0<br>$2.0 + \frac{h}{2}$ | maximum delay value, droop rounding rule<br>maximum delay value, equal rounding rule, resolution $h$ |

## 2.3 Microcircuit Network

The second network model we will use to explore the differences in spiking activity based on the choices of the delay values of the neurons, is the microcircuit model by Potjans and Diesmann (2012). The model is an extension of the balanced random network model. It is built up of leaky integrate-and-fire neurons having static synapses where the inhibitory synapses are stronger than the excitatory ones to achieve a balanced network. The connectivity and external inputs in the model have been found by integrating two connectivity maps, from anatomy and physiology. The integrated connectivity map have been found by comparing them to experimental data from several species with a focus on the rat brain and the cat brain. For further details on how the model was developed, see the article Potjans and Diesmann (2012).

The microcircuit model is supposed to present a network in the cortex under a 1mm$^2$ surface area. This network is presented by four layers, $L2/3$, $L4$, $L5$ and $L6$. In each layer we have a population of excitatory and a population of inhibitory neurons. The neuron populations are connected either through "within-layer" connections, that is, the populations lay in the same layer. Or they are connected through "intralayer" connections, that is, the populations lie in different layers. In addition to the input neurons get from neurons in the other populations in the network, they also receive background noise in the form of Poisson spike trains. In addition to the background Poissonian input to each layer, the layers $L4$ and $L6$ also receive thalamo-cortical input.

As was the case for the Brunel network, Sec. 2.2, the neurons in this network are represented as leaky integrate-and-fire neurons, Sec. 2.1. In this network, the neurons have synaptic current inputs that are exponentially shaped (Potjans and Diesmann 2012). That is, the synaptic input current can be described through the equation (Sterratt et al. 2011, Ch. 8)

$$I_{syn}(t) = \begin{cases} \bar{I}_{syn} exp(-\frac{t-t_s}{\tau_{syn}}) & \text{for } t \geq t_s \\ 0 & \text{for } t < t_s \end{cases} \tag{2.6}$$

In the equation, $t_s$ describes the arrival of a spike, which can be described as the time of emission of a spike from the connecting neuron plus the delay.

The neurons are randomly connected according to the probabilities in the connectivity map. The delays of the neurons are drawn from a normal distribution with mean 1.5ms and standard deviation 0.75ms for the excitatory synapses and mean 0.8ms and standard deviation 0.4ms for the inhibitory synapses. There are about 80000 neurons in the network with 300 million synapses. The synaptic strengths are drawn from a Gaussian distribution. The strength of the inhibitory synapses arriving at each neuron, or postsynaptic potential, are negative and increased by a factor of $g$ compared to the excitatory ones (Potjans and Diesmann 2012).

We give an overview of the network model and the parameters used in our simulations in the following tables, which are similar to those in the thesis by Dasbach (2020) written in tables after Nordlie et al. (2009).

| A | Model Summary |
|---|---|
| **Populations** | 8 cortical populations in 4 layers (L2/3E, L2/3I, L4E, L4I, L5E, L5I, L6E, L6I) |
| **Topology** | no spatial information |
| **Connectivity** | Random connections, population-specific, fixed number of synapses |
| **Neuron model** | Leaky integrate-and-fire, fixed voltage threshold, fixed absolute refractory time (voltage clamp) |
| **Synapse model** | Exponential-shaped postsynaptic currents with static, normally distributed weights |
| **Input** | No input used |
| **Measurements** | Spike times |

| B | Network model |
|---|---|
| **Connectivity** | Fixed number of synapses $S_{YX}$ between populations $X$ and $Y$. Connection probabilities $C_{YX}$ from population $X$ to population $Y$ with $X, Y \in \{L2/3, L4, L5, L6\} \times \{E, I\}$ is $C_{YX} = 1 - \left(1 - \frac{1}{N_{pre}N_{post}}\right)^{S_{YX}}$ |

| C | Neuron and Synapse Model |
|---|---|
| **Name** | Iaf neuron |
| **Type** | Leaky integrate-and-fire, exponential current input |
| **Subthreshold dynamics** | $\tau \frac{dV}{dt} = -V(t) + RI(t)$ if $t \notin [t_i^k, t_i^k + \tau_{\mathrm{rp}})$<br>$V(t) = V_{\mathrm{r}}$ else<br>$V(t=0) \sim \mathcal{N}(\mu = V_{0\mu}, \sigma^2 = V_{0\sigma})$ |
| **Spiking** | The neuron spikes at time $t_i^k$ if $V(t_i^k) \geq V_\theta$. |
| **Synapse current** | $I_i(t) = \sum_j J_{ij} \sum_k e^{-(t-t_j^k-d_{ij})/\tau_s} \Theta(t - t_j^k - d_{ij})$,<br>where $\Theta(t) = \begin{cases} 1 & \text{if } t \geq 0 \\ 0 & \text{else} \end{cases}$<br>$J_{ij} \sim \mathcal{N}(\mu = g_{YX} J, \sigma^2 = \sigma^2_{J,YX})$<br>Delay values are drawn from uniform distributions [min,max] for droop rounding rule and $[\min - \frac{h}{2}, \max + \frac{h}{2}]$ for equal rounding rule with resolution $h$, min=$\mu_{\mathrm{exc(inh)}} - \mu_{\mathrm{exc(inh)}} \cdot \sigma$, max=$\mu_{\mathrm{exc(inh)}} + \mu_{\mathrm{exc(inh)}} \cdot \sigma$. |

| D | Populations | |
|---|---|---|
| **Name** | **Elements** | **Size** |
| L2/3E | Iaf neuron | 20683 |
| L2/3I | Iaf neuron | 5834 |
| L4E | Iaf neuron | 21915 |
| L4I | Iaf neuron | 5479 |
| L5E | Iaf neuron | 4850 |
| L5I | Iaf neuron | 1065 |
| L6E | Iaf neuron | 14395 |
| L6I | Iaf neuron | 2948 |

| E | | Parameters |
|---|---|---|
| **Name** | **Value** | **Description** |
| $J$ | 0.15 mV | reference synaptic strength |
| $g_{XY}$ | 1 | $X \in \{L5E, L6E\}$ |
| | $-4$ | $X \in \{L2/3I, L4I, L5I, L6I\}$ |
| | 2 | $(X, Y) = \{L4E, L2/3E\}$ |
| $\sigma_{J,XY}$ | $0.1 \cdot J \cdot g_{XY}$ | standard deviation of weight distribution |
| $C_m$ | 250 pF | membrane capacitance |
| $\tau_m$ | 10 ms | membrane time constant |
| $\tau_{ref}$ | 2 ms | absolute refractory period |
| $V_r$ | $-65$ mV | reset potential |
| $V_\theta$ | $-50$ mV | spike threshold |
| $E_L$ | $-65$ mV | leak reversal potential of neurons |
| $\tau_s$ | 0.5 ms | Postsynaptic current time constant |
| $\mu_{\text{exc}}$ | 1.5 | mean excitatory delay |
| $\mu_{\text{inh}}$ | 0.75 | mean inhibitory delay |
| $\sigma$ | 0.5 | standard deviation of delay values |
| $V_{0\mu}$ | $-68.28$ mV | Mean starting potential population L2/3E |
| | $-63.16$ mV | L2/3I |
| | $-63.33$ mV | L4E |
| | $-63.45$ mV | L4I |
| | $-63.11$ mV | L5E |
| | $-61.66$ mV | L5I |
| | $-66.72$ mV | L6E |
| | $-61.43$ mV | L6I |
| $V_{0\sigma}$ | 5.36 mV | Standard deviation of starting potential population L2/3E |
| | 4.57 mV | L2/3I |
| | 4.74 mV | L4E |
| | 4.94 mV | L4I |
| | 4.94 mV | L5E |
| | 4.55 mV | L5I |
| | 5.46 mV | L6E |
| | 4.48 mV | L6I |
| $T$ | 10 s | Simulation duration |

| **F** | | **Connection probabilities $C_{YX}$** | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | From $X$ | | | | | | | |
| | | L2/3E | L2/3I | L4E | L4I | L5E | L5I | L6E | L6I |
| To Y | L2/3E | 0.1009 | 0.1689 | 0.0437 | 0.0818 | 0.0323 | 0 | 0.0076 | 0 |
| | L2/3I | 0.1346 | 0.1371 | 0.0316 | 0.0515 | 0.0755 | 0 | 0.0042 | 0 |
| | L4E | 0.0077 | 0.0059 | 0.0497 | 0.135 | 0.0067 | 0.0003 | 0.0453 | 0 |
| | L4I | 0.0691 | 0.0029 | 0.0794 | 0.1597 | 0.0033 | 0 | 0.1057 | 0 |
| | L5E | 0.1004 | 0.0622 | 0.0505 | 0.0057 | 0.0831 | 0.3726 | 0.0204 | 0 |
| | L5I | 0.0548 | 0.0269 | 0.0257 | 0.0022 | 0.06 | 0.3158 | 0.0086 | 0 |
| | L6E | 0.0156 | 0.0066 | 0.0211 | 0.0166 | 0.0572 | 0.0197 | 0.0396 | 0.2252 |
| | L6I | 0.0364 | 0.001 | 0.0034 | 0.0005 | 0.0277 | 0.008 | 0.0658 | 0.1443 |

## 2.4 Statistics

In order to see the effects of how the delays of spikes sent between neurons are drawn, several statistics will be considered. The precise spiking times of a neural network will differ for each simulation, even when the same network is simulated. It is therefore more useful to look at the overall statistics of spiking activity. To compare spiking activity, we will look into the distributions of three spiking statistics, the firing rate ($FR$), the coefficient of variation ($CV$), and the Pearson correlation coefficient ($CC$). The idea of using these statistics to compare spiking activity came from the master thesis work of Dasbach (2020).

To get an overall overview of the activity of the neurons, one can calculate the mean firing rate of each neuron $i$. It is calculated as

$$FR_i = \frac{S_i}{T} \ . \tag{2.7}$$

In this equation, $S_i$ refers to the number of spikes emitted from neuron $i$ during the time period $[t, t + \Delta t]$. $T$ refers to the length of the time period for which the number of spikes were counted (Sterratt et al. 2011). That is, the mean firing rate of a neuron is given in spikes per second. We get a $FR$ distribution for each model, which consists of the $FR$ value of each neuron in the network.

Another useful statistic is the coefficient of variation, which can be calculated for each neuron in the network. It gives a quick measure of how irregular the firing of the neuron is. The closer to zero we get, the more regular the firing is. It can be calculated using the interspike intervals (ISIs). An interspike interval is the time between two consecutive spikes fired from a neuron. We find all interspike intervals of a neuron, then the coefficient of variation can be calculated as

$$CV = \frac{\sigma_{\mathrm{ISI}}}{\mu_{\mathrm{ISI}}} \; . \tag{2.8}$$

$\mu_{\mathrm{ISI}}$ is the mean of the interspike intervals, and $\sigma_{\mathrm{ISI}}$ is the standard deviation of the interspike intervals (Gabbiani and Koch 1998). The $CV$ is calculated for each neuron in the model. Then we find the $CV$ distribution of a model which is simply the $CV$ value of each neuron in the network.

Then, to get an overview of the correlation between the firing in one neuron and the other, we use the Pearson correlation coefficient (Devore and Berk 2012). It is defined for two distributions $X$ and $Y$ as

$$r = \frac{\sum(x_i - \bar{x})(y_j - \bar{y})}{\sqrt{\sum(x_i - \bar{x})^2 \sum(y_j - \bar{y})^2}}$$

If we apply it to neurons $i$ and $j$ it can be calculated as

$$CC_{i,j} = \frac{\sum(t_i - \mu_i)(t_j - \mu_j)}{\sqrt{\sum(t_i - \mu_i)^2 \sum(t_j - \mu_j)^2}} \; . \tag{2.9}$$

$t_i$ and $t_j$ represents the spike times of neurons $i$ and $j$, and $\mu_i$ and $\mu_j$ represents the mean of the spike trains of neuron $i$ and $j$, respectively. In contrast to the mean firing rate which is given in spikes per second, the coefficient of variation and the Pearson correlation coefficient do not have units.

In the case of the Brunel network model, we choose to calculate the correlation between each excitatory neuron in the network with the first inhibitory neuron. As there are recordings of 12500 neurons from each simulation, calculating the $CC$ for each possible pair of neurons would not provide much more information. In the case of the Microcircuit network model, we look into the spiking of population L5I. We calculate the correlation between each neuron in the population and the first neuron of that same population. Choosing to compute the correlations between each neuron with a single neuron is convenient as we get the same number of values for the $CC$ distributions as we get for the $CV$ and $FR$ distributions. This in turn, makes the organization of all the data into a single data frame feasible.

To compare the models, it is of interest to use some sort of distance measure. The first thing that comes to mind is the Kolmogorov-Smirnov test (Conover 1999). The Kolmogorov-Smirnov test calculates the empirical distribution functions of the two samples being compared and finds their maximum difference, given as the maximum vertical distance between the empirical distribution functions. The test then accepts the null hypothesis of the two samples being from the same distribution if the maximum distance is small or the p-value is high. However when this test is performed on simulations from the same model with the same resolution, only having a different seed value, the test does not pass for many cases. The test can therefore not be used as it does not correctly pass when two simulations are from the same model.

Another idea could be to use the Kullback-Leibler (KL) divergence, which measures how different two density distributions are. According to Joyce (2011), it is however not a genuine metric, as it is not symmetric and fails the triangle inequality. It is therefore not clear how to interpret this difference in terms of deciding which models are closer. Should the KL divergence between two models be closer to zero for them to be more equal, or should it be closer to the KL divergence values of the reference model against itself? As it is not clear what the KL divergence actually tells us about the models, we decide to not proceed with this measure.

The Wasserstein distance metric, however, can be used to compare the various models against each other. If we let $P$ and $Q$ denote two probability measures on $\mathbb{R}^d$, then according to Ramdas et al. (2017), the Wasserstein distance between them is calculated as

$$W_p(P,Q) = \left( \inf_{\pi \in \Gamma(P,Q)} \int_{\mathbb{R}^d \times \mathbb{R}^d} ||X - Y||^p d\pi \right)^{\frac{1}{p}} . \tag{2.10}$$

According to Ramdas et al. (2017), $\Gamma(P,Q)$ is the set of all joint probability measures on $\mathbb{R}^d \times \mathbb{R}^d$ whose marginals are $P$ and $Q$.

Intuitively one can view the Wasserstein distance as a measure of the cost of turning one probability density function into another. In our results, we will present the Wasserstein distance calculated based on the probability density functions of the $FR$, $CV$ and $CC$ values of the different network models and cases. The $CC$ values will be calculated between each possible pair of neurons in the network for this purpose.

To get an understanding of what is viewed as a small Wasserstein distance, the distance will be calculated between the first seeded simulation of the continuous model against all the other seeded simulations of the same model. For the discrete models, for each rounding rule and each resolution, we calculate the distance for all ten seeded simulations against the first seeded simulation of the continuous model. We then look at the mean of these ten distances to compare the models and decide on which one is closest to the reference model.

# Chapter 3

# Methods

## 3.1 Delay values

It is in our interest to study the effects the rounding of the delay values have on the spiking activity in our network models. We will draw the delay values from a uniformly distributed interval with three different rounding rules. A continuous version where the delay values are not rounded, they are simply just taken from the continuous uniformly distributed interval. Then we have a droop version, in this case the delay values are rounded to the nearest grid point, making the minimum and maximum value of the interval have only half the probability of being drawn. The minimum value can only be rounded down to from one side, and the maximum value can only be rounded up to from one side, while the points in between can be rounded up and down to from both sides. The third rounding rule we explore is named the equal model. In this case we also round the spike times, but we add values to the interval on both sides to make sure the minimum and maximum value have the same probability of being drawn as the other values do.

To make this more clear, see the illustration for the two rounding rules in the case of resolution $\frac{1}{8}$ when the delay values are drawn from the interval $[1.0, 2.0]$ in Figure 3.1. In the figure one can see how the interval $[1.0, 2.0]$ is split into eight intervals as the resolution is $\frac{1}{8}$. Then all values falling into the red colored sections will be rounded to the value marked with a point in that section. Note how for the droop rounding rule, displayed in the top of the figure, the red colored area is only half as long for the minimum and maximum value, 1.0 and 2.0. For the equal rounding rule, displayed in the bottom of the figure, the red colored section have the same length for all grid point values. In the case of the equal model the delays are drawn from the interval $[0.9375, 2.0625]$ which is $[1.0-\frac{h}{2}, 2.0+\frac{h}{2}]$ where $h$ denotes the resolution $\frac{1}{8}$.

For the Brunel network we are looking at delay values drawn from the uniform distribution on $[1.0, 2.0]$ for the droop model and the continuous model.

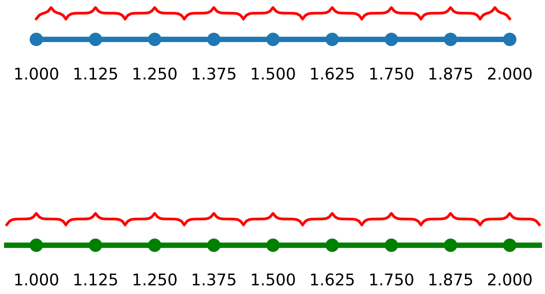


**Figure 3.1:** Delay values drawn from interval $[1.0, 2.0]$ for the droop model with resolution $\frac{1}{8}$ on top, and delay values drawn from interval $[0.9375, 2.0625]$ for the equal model with resolution $\frac{1}{8}$ on the bottom. The numbers falling within each area marked by red brackets will be rounded to the grid point marked by a point on the line within that area.

While for the equal model the delays are drawn from $[1.0 - \frac{h}{2}, 2.0 + \frac{h}{2}]$. For the microcircuit network, each delay value is drawn from the uniform distribution $[\mu - \mu\sigma, \mu + \mu\sigma]$ for the droop and the continuous model. Where $\sigma = 0.5$ and $\mu = 1.5$ for excitatory and $\mu = 0.75$ for inhibitory synapse delays. For the equal model the delays are drawn from the interval $[\mu - \mu\sigma - h/2, \mu + \mu\sigma + h/2]$ where $h$ is the resolution.

So far we have discussed three rounding rules/models. The continuous model where the spike times and the delay values are continuous. The droop model where the spike times and the delay values are discrete, with unequal probability of drawing the delay values at each end of the interval. The third model was the equal model where the spike times and the delay values are discrete, but the probability of the delay values are all equal. However, it could be of interest to see how the discretization of only the delay values affect the spiking activity. We will therefore also examine a version of the network models where the spike times are continuous, but the delay values are drawn from the discrete uniform distribution. We name this version the “semi-continuous” model.

## 3.2 Generating data

In this thesis, the spiking data was generated using several tools. The simulation of a network is created using the NEST simulator (Deepu et al. 2021). NEST focuses on recreating spiking activity in large networks, rather than focusing on recreating exact models of single neurons. It is therefore useful for trying to recreate and understand dynamics of larger networks. As NEST is originally implemented in SLI, a language derived from Postscript (Adobe Systems Incorporated 1999), an interface called PyNEST (Eppler et al. 2009)

| | |
|---|---|
| CPU | 2 x AMD EPYC 7742, 2 x 64 cores, 2.25GHz |
| RAM | 256 (16 x 16) GB DDR4, 3200MHz |
| Ethernet connection | InfiniBand HDR100 (Connect X6) |
| Storage | 1× 240 GB SSD (user files), 1TB NVMe (program files) |

**Table 3.1:** JUSUF compute node configuration.

has been used. It forwards commands from Python to the NEST simulation kernel via SLI, making it possible to run simulations via Python (Van Rossum and Drake 2009). To get general and representative data of neuronal spiking, the simulations have been performed multiple times with only a few variables alternating. Such as the seed values and the minimum and maximum delay values. To effectively run the simulations with alternating variables, the beNNch framework (Albers et al. 2021) was used. beNNch also offers analysis and results functionalities, giving analysis of e.g. the computational power needed to perform the simulations.

The networks are quite large in size and are simulated over long periods of time. In some cases, running them on a local computer is not feasible and hence, supercomputers were used. The simulations were run on the supercomputer JUSUF at the Jülich research center. The configuration can be found under Jülich Supercomputing Center (2021). It consists of 144 standard compute nodes, which has the specifications summarized in Table 3.1.

The spikes emitted from a neural network will vary for each run, even if the same type of network is used. To get an overall representation of the spiking activity, the same network is simulated for a longer period of time, about 10$s$, for different choices of seed value. That is, the seed used by the NEST simulator when generating the random numbers. These random numbers occur first in the noise generators connected to the neurons, and then the randomness at which the delay values are drawn when the neurons are getting connected to each other. To make sure that our results and conclusions are not built on a specific case of each model, where e.g. a specific choice of seed value made the droop model closer to the reference model than the equal model, we make sure to run the simulations for several choices of seed value.

In addition to the randomness of the noise and delay values, the resolution at which the simulations are run plays a part. The difference between models where continuous versus discrete values are used will be strongly affected by the resolution as fewer values will be drawn for the discrete cases with low resolutions. We do however want to find the model with the lowest possible computational cost that is still presentable, in that the spiking activity is still similar to that of the reference model. Because of this, the simulations are also

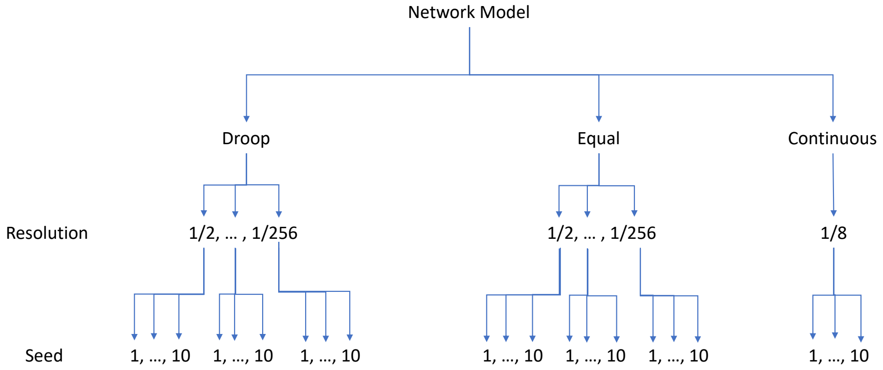


**Figure 3.2:** Setup of simulations for a single network. The two different rounding rules version of the network is run for eight resolutions, and each resolution is again run for ten different seed values. The continuous version is only ran for one resolution with ten different seed values.

run for different resolutions. The ten different simulations of the same model, each with its own seed value, were run for resolutions $\frac{1}{2}, \cdots, \frac{1}{2^n}, \cdots, \frac{1}{256}$. The resolutions that were used are all a factor of $\frac{1}{2}$ to avoid rounding errors in the binary system.

The ten differently seeded simulations with the eight different resolutions is done for two different models. The droop model and the equal model. To get a reference point, the continuous model is simulated. This model is viewed as the reference point, or the reference model. It is only run for one resolution, with ten differently seeded simulations. Whether using the droop or the equal model is more correct can be determined by examining which model converges faster towards the reference model. To make it more clear, the setup of simulations for a single network is illustrated in Figure 3.2.

We are comparing 10000 excitatory neurons in the case of the Brunel model, and 1065 inhibitory neurons for the microcircuit model, for $10000ms$, for ten different seed values, in three different models, for eight different resolutions. We end up having 1.7million spike trains over 10000ms for the Brunel network, and around 1.4million spike trains for the Microcircuit network. Getting an overview and determining whether models perform similar based on these data points becomes complex if we simply look at the spike times and raster plots of these. To make an analysis easier we instead look into the statistics of the spike times. To perform the analysis of the models, a data set consisting of the $CV$, $FR$ and $CC$ distributions for each possible version of each model is used. That is, for each rounding rule, for each resolution and each seed value, the distributions are found.

In order to calculate the statistics ($FR$, $CV$, $CC$) of the spiking activity, the python packages neo (Garcia et al. 2014) and elephant (Denker et al.

2018) are used. The package neo is used to create “SpikeTrain” objects of our spike times given in milliseconds, and then elephant is used to calculate the statistics of these objects. In order to calculate Wasserstein distances between the models based on their statistics, the NetworkUnit (Gutzen et al. 2018) package is used. It is a module built on the SciUnit package and can be used to validate neuronal network models against each other through tests.

# Chapter 4

# Results

We will now introduce the results of our simulations. We start by displaying the results of the Brunel network, both the statistics and the distance measure defined in Sec. 2.4. Then we go on to introduce the corresponding results for the Microcircuit network. We have chosen to show the results for the excitatory population in the Brunel network, and the inhibitory neurons in population $L5I$ for the Microcircuit network. In addition to the standard Brunel network, we will also examine the statistics of the Brunel network model where only the delays are discretized while the spike times are continuous. Lastly we include the run times for the network models. That way one can take computational cost into consideration when comparing the models and their similarities.

## 4.1 Brunel Network

We want to display the mean distributions of $FR$, $CV$ and $CC$ (Sec. 2.4). They will be displayed by empirical cumulative density functions (ecdf). By mean distribution, we refer to a distribution consisting of the mean statistic of each neuron, taken across the ten seeded simulations. That is, for e.g. the droop model run with resolution $\frac{1}{8}$, we calculate for each of the 10000 neurons, the FR value in each of the ten seeded simulations. Then for each neuron we take the mean of these ten FR values and view the 10000 means as a distribution. See an example of how this looks in the plots of Figure 4.1. In the figure on the left hand side, we have eight ecdf curves, each representing one seeded simulation of the droop model run with resolution $\frac{1}{8}$. Each curve represents the CV value of each of the 10000 neurons in the network. The plot in the figure on the right hand side shows one ecdf curve, displaying the mean CV distribution of the droop model run with resolution $\frac{1}{8}$. This curve represents 10000 mean CV values. Where each mean CV value is the mean of ten different CV values for the same neuron.

In addition to the ecdf curves, we want to display the overall mean statistics as a function of resolution. For each of the droop, equal and continuous model,

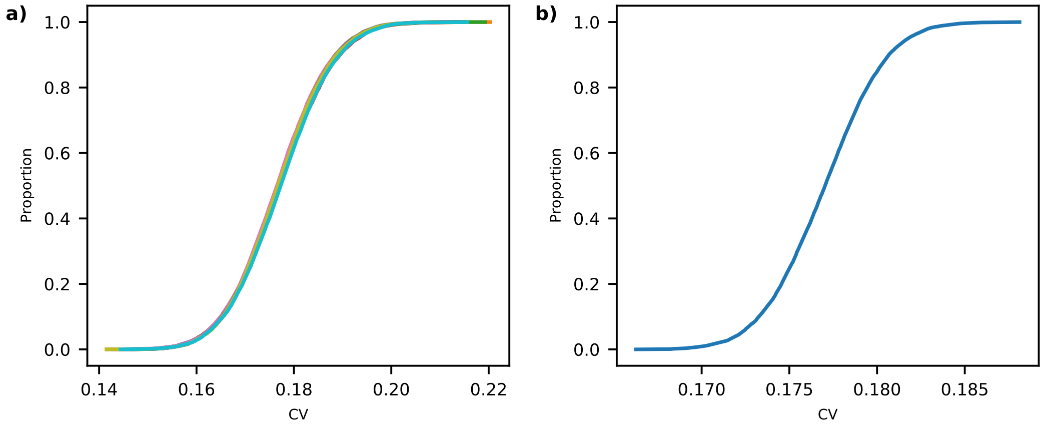


**Figure 4.1:** The ecdf curves of the coefficient of variation for the droop model with resolution 1/8. An ecdf curve for each seeded simulation displayed on the left, and a single ecdf curve displaying the mean distribution of the seeded simulations displayed on the right.

for each resolution, for each seeded simulation, we calculate the statistic of each neuron in the population and take the mean of all the neurons statistics. After this is done for all ten seeded simulation, the mean and the standard deviation of these ten means is taken. The means will be used to present the model (droop, equal or continuous) for that resolution. They will be displayed in figures together with the standard deviations to get an idea of how the statistics vary.

To examine the results of using the different rounding rules in delay values on spiking activity in the Brunel network, we examine the mean statistics shown in Figure 4.2. On the left hand side we can see empirical cumulative density function plots. With a figure for each of the three statistics: firing rate, coefficient of variation and the Pearson correlation coefficient. There is a single curve for one resolution, one type of model. That is, for each of the droop and the equal model, there are eight curves, showing the mean distribution taken over the ten seeded simulations for that case. There is a single curve showing the continuous model, as it was simulated for one resolution. The equal model is displayed with dashed lines, and the color of the droop and the equal model are the same for the same resolution. E.g. for resolution $\frac{1}{2}$ we have a red-ish purple color, the dashed line is for the equal model while the solid line is for the droop model. The colors become gradually lighter until it is light green. The lighter the color is the higher the resolution of the network simulations are. That is, the lightest green shows the statistics as the resolution is $\frac{1}{256}$ and the continuous model is represented with a yellow color. One can see which color represents which resolution in the legend in the top left plot of Figure 4.2.

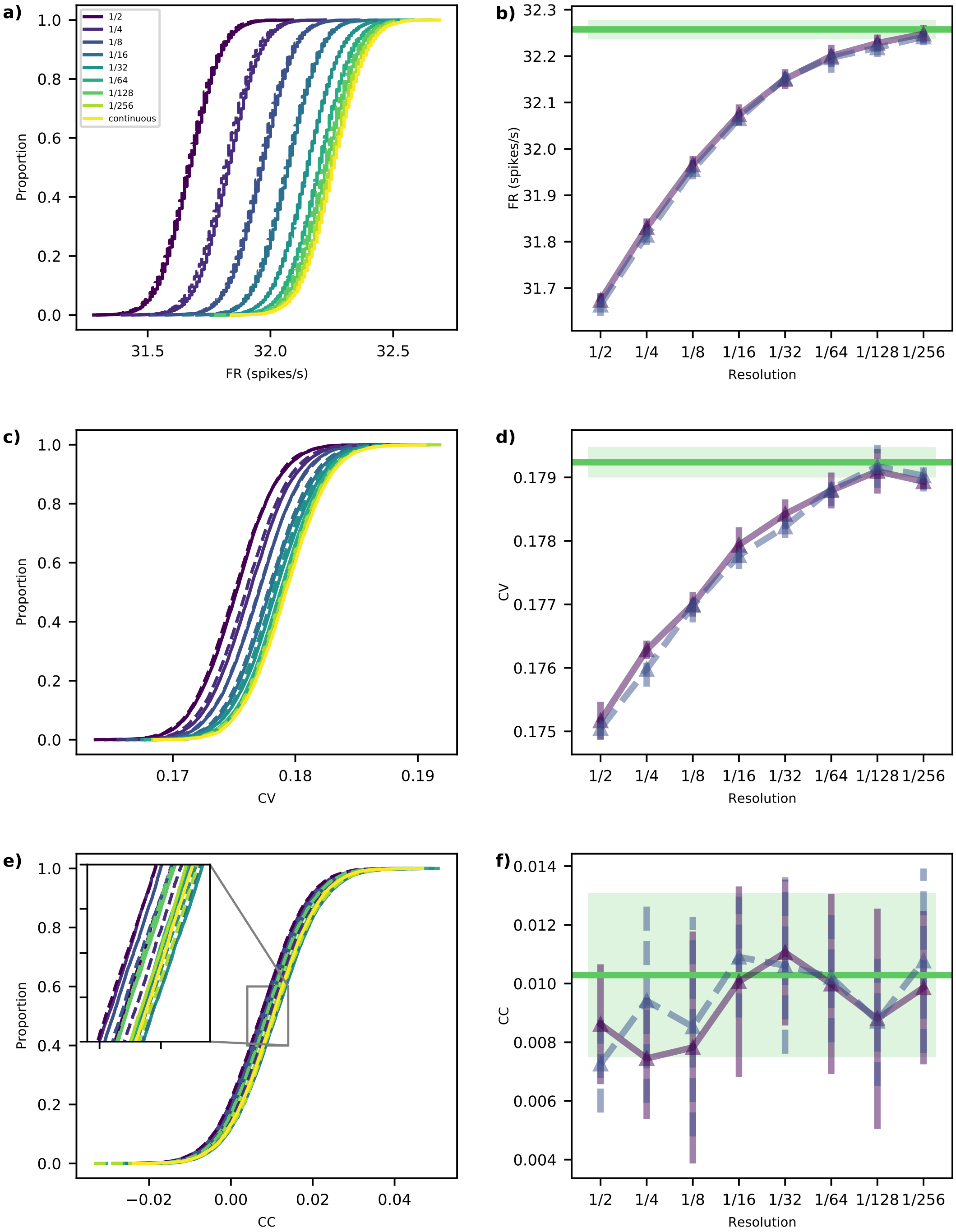


**Figure 4.2:** From the top: mean FR, CV and CC distributions for all three cases, droop, equal and continuous. Both as ecdf curves and as functions of resolution. The ecdf curves on the left shows the equal model in dashed lines, the equal and the droop model have the same color for the same resolutions. The yellow curve represents the continuous case. On the right hand side, the droop model is represented as purple, the equal model as dashed blue lines, and the continuous model in green. The vertical lines represents the variation in the values. The shaded green area shows the variation of the continuous model's mean value.

The ecdf curves representing the mean FR distributions in the top left of Figure 4.2, are quite easy to distinguish. In this plot the curves of the equal and the droop model lay close together when they have the same resolution. This can be seen for e.g. resolution $\frac{1}{4}$ where the droop and the equal model's curve have a difference of around 0.016 spikes/s, while the difference between the equal model for resolutions $\frac{1}{2}$ and $\frac{1}{4}$ is about 0.14 spikes/s. As the resolution increases, the curves representing the droop and the equal model comes closer to the curve representing the continuous model. When the resolution is at its highest, $\frac{1}{256}$, the curves of the equal and the droop model are also at their closest to the curve for the continuous model. The difference between the curves of the equal and the droop model from the curve of the continuous model is around 0.6 spikes/s for the lowest resolution, while for the highest resolution the difference is around 0.016 spikes/s for the equal model and 0.0095 spikes/s for the droop model.

When examining the mean CV distribution of the models shown in the middle left plot of Figure 4.2, one can distinguish the curves to some degree, however they are more gathered compared to the case of the FR distributions. The curves for the equal and the droop model seem to lie close together when they represent the same resolution. E.g. the difference between the curves of the two lowest resolutions for the equal model is around 0.0009, while the difference between the equal and the droop models' curves for the lowest resolution is around 0.00012. One also notices that the higher the resolution becomes, the more the curves shift to the right, which is closer to the curve of the continuous model. For the lower resolutions, the droop model, displayed with solid lines, is slightly closer to the curve of the continuous model. One can see this as the solid lines lay further to the right than the dashed lines, representing the equal model. But as the resolution increases the possible differences become harder to see.

In the case of the CC distributions, displayed in the bottom left plot of Figure 4.2, the differences between the curves representing the different resolutions is not that clear. Most of the curves lay on top of each other, closely gathered around the one representing the continuous model. One can get a better view of which curves lay close together when examining the zoomed in plot of the curves. The curves representing the higher resolutions seem to lay the closest to the curve of the continuous model, with an exception of resolution $\frac{1}{128}$ for both the droop and the equal model.

To get a better overview of the mean statistics of the two discrete models and the continuous model, and thereby a better overview of the differences between them, we examine the mean statistics as a function of resolution. They are displayed in the figures on the right hand side of Figure 4.2. The purple line represents the droop model, the dashed blue line represents the equal model and the green line is the continuous model's mean statistic. In addition to the mean values, there are vertical lines representing the variation in the mean of the statistics. They have the same color as the lines representing the mean. The shaded green area shows the variation of the mean statistic of the continuous model.

We start by examining the mean FR values displayed in the top right plot of Figure 4.2. In this case the droop and the equal models mean values are quite different from the mean FR value of the continuous model for lower resolutions. E.g. for resolution $\frac{1}{2}$ the difference between the droop and the equal models mean from the continuous model's mean is about 0.6 spikes/s, while for resolutions higher than $\frac{1}{64}$ this difference is less than 0.06 spikes/s. As the resolution increases the differences between the discrete models and the continuous model's mean FR values get smaller. For the highest resolution the mean FR values of the discrete models are at its closest, with a difference of 0.0095 spikes/s for the droop model, and a difference of 0.016 spikes/s for the equal model. However, when it comes to distinguishing the mean FR values of the droop and the equal model, the difference is not that clear. The lines are almost completely overlapping. There is a maximum difference between the equal and the droop models mean FR of 0.016 spikes/s. It might look like the mean FR of the droop model is slightly closer to the mean FR of the continuous model for lower resolutions ($\frac{1}{2}, \dots, \frac{1}{16}$), and for the highest resolutions ($\frac{1}{128}, \frac{1}{256}$). However, as mentioned, the maximum difference in the mean FR values for the droop and the equal model is 0.016 spikes/s, while the standard deviations are of 0.02 spikes/s.

The mean CV values are displayed in the middle right plot of Figure 4.2. For all resolutions the droop and the equal models have a difference between their mean CV values of less than 0.0003. The droop models mean CV is slightly less different to the mean CV of the continuous model for the lower resolutions, up until resolution $\frac{1}{64}$. Then for higher resolutions, the equal model seems to have a less different mean CV value compared to the continuous model's mean. The bars representing the variances of the discrete models means are overlapping for the most part, and has a size of 0.0002. As the resolution increases, the lines representing the mean CV values of the droop and the equal model gets closer to the line representing the mean CV value of the continuous model. For e.g. resolution $\frac{1}{2}$ the difference between the mean CV values of the droop and the equal model from the continuous model's mean is about 0.004, while for resolution $\frac{1}{64}$ the difference is about 0.0004. For resolution $\frac{1}{128}$ the mean CV values of the droop and the equal model are within the range of standard deviation from the mean of the continuous model's CV values, represented through the shaded green area.

Lastly we examine the mean CC values of the three models, displayed in the bottom right of Figure 4.2. There is no clear difference between the models mean values in the sense of having the least difference in mean value from the continuous model's mean. The difference in mean CC values of the equal and the droop model from the continuous model is less than 0.003 already from the low resolutions, while the standard deviation is of size 0.003, approximately. They are slightly changing in getting larger and smaller differences from the continuous model's mean as the resolution changes. The equal model has a smaller difference in the mean CC value from the continuous model's mean for some resolutions ($\frac{1}{4}$, $\frac{1}{8}$, $\frac{1}{256}$). While for other resolutions the droop models mean CC value has less difference from the continuous model's mean ($\frac{1}{2}$, $\frac{1}{16}$). The droop model has mean value with least difference from the mean of the continuous model for resolution $\frac{1}{16}$, with a difference of approximately 0.0002. The mean of the equal model however, has the least difference for resolution $\frac{1}{64}$ with a difference of approximately 0.00007. Note that the standard deviations have a size of 0.003.

To get a more detailed overview of the mean statistics as a function of resolution, we include the numbers in Table 4.1.

In order to see if maybe highly irregular neuronal activity would uncover a trend of one of the discrete models being more similar to the continuous one, the Brunel network was simulated with strong synapses. We simulated the network with the parameter $J$, representing the synaptic strengths, increased by a factor of 10. Increasing the synaptic strength ensures that we get highly irregular individual neuronal activity (Kriener et al. 2014). We did see an increase in the CV values, meaning we have more irregular spiking. There were also some changes in the values of the other statistics. One could see some differences in which of the droop and the equal models' mean values was the least different from the mean of the continuous model for a specific resolution. However, also in this case, the discrete model having the least different mean from the continuous model's mean value changed as resolution changed. There was no trend of one model being more similar to the continuous model in regards to the mean values of the statistics. Therefore, we do not include the statistics here as no new insights were given from these simulations.

### 4.1.1 Wasserstein Distances

To make the differences between the models mean statistics even more clear, we also display the mean Wasserstein distances. The Wasserstein distance metric is calculated based on the $FR$, $CV$ and $CC$ values of the models. The mean distances are displayed in Figure 4.3, where the green line represents the Wasserstein distance between the first seeded simulation of the continuous model and all other seeded simulations in the continuous model. We can use the green line as a reference point of what we consider short distances. The purple solid curve represents the distance between the droop model and the first seeded simulation of the continuous model, while the dashed blue curve represents the distance between the equal model and the first seeded simulation of the continuous model.

The Wasserstein distances calculated based on the firing rates are displayed

| | | Resolutions | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | 1/2 | 1/4 | 1/8 | 1/16 | 1/32 | 1/64 | 1/128 | 1/256 |
| Droop | FR (spikes/s) | 31.674 | 31.831 | 31.966 | 32.074 | 32.150 | 32.200 | 32.228 | 32.248 |
| | CV | 0.1751 | 0.1763 | 0.1770 | 0.1779 | 0.1784 | 0.1788 | 0.1791 | 0.1789 |
| | CC | 0.00861 | 0.00745 | 0.00782 | 0.01006 | 0.01106 | 0.00999 | 0.00880 | 0.00986 |
| Equal | FR (spikes/s) | 31.664 | 31.814 | 31.955 | 32.065 | 32.152 | 32.197 | 32.218 | 32.242 |
| | CV | 0.1750 | 0.1760 | 0.1770 | 0.1778 | 0.1782 | 0.1788 | 0.1792 | 0.1790 |
| | CC | 0.00726 | 0.00941 | 0.00853 | 0.01090 | 0.01062 | 0.01022 | 0.00872 | 0.01077 |
| Continuous | FR (spikes/s) | - | - | 32.257 | - | - | - | - | - |
| | CV | - | - | 0.1792 | - | - | - | - | - |
| | CC | - | - | 0.01029 | - | - | - | - | - |

**Table 4.1:** The mean FR, CV and CC value of the droop, equal and continuous model for all resolutions in the Brunel network.

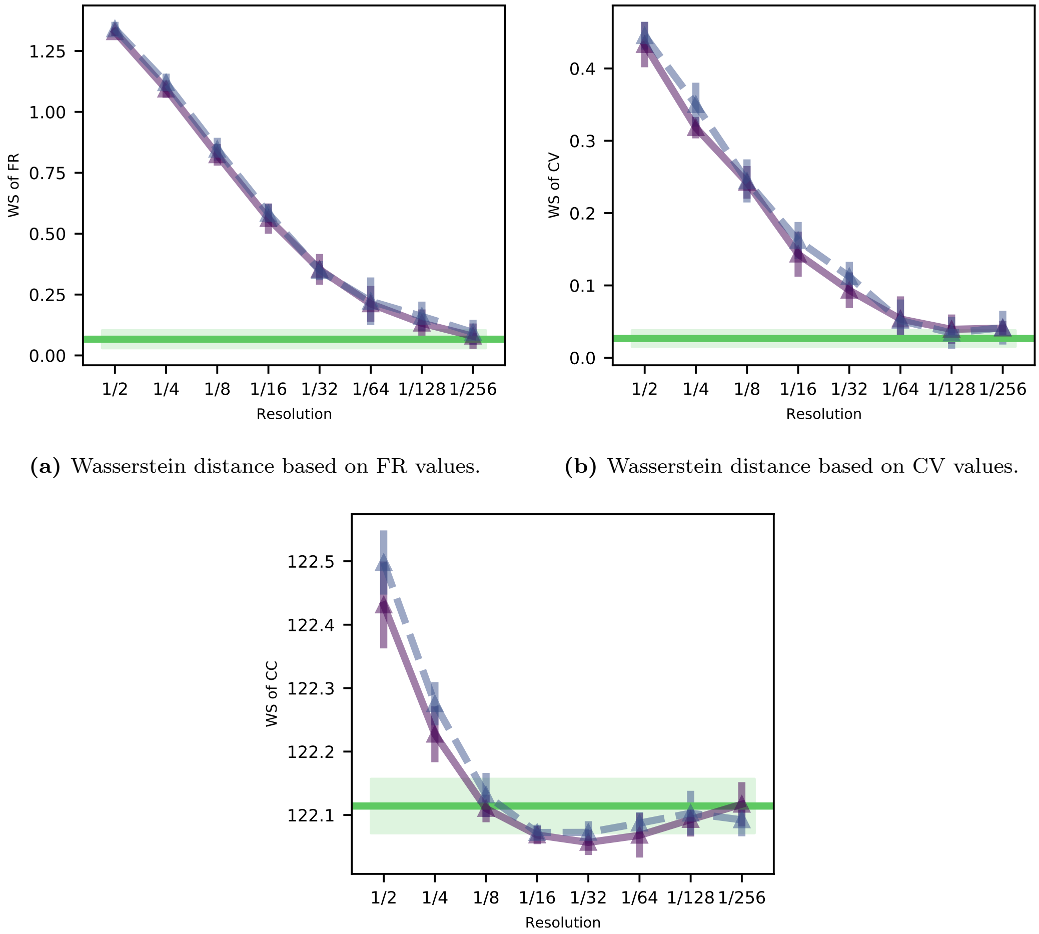
in Figure 4.3a. We can see how the distances for the droop and the equal model become smaller as resolution increases. As we get to the highest resolution, the distances of the discrete models are only differing from the mean reference distance by 0.03 while the mean of the reference distances has a standard deviation of 0.037. That is, they fall within the range of distances seen as small. The distances of the two discrete models seem to be similar for all resolutions, differing by maximum 0.03. For each resolution the distance for the droop model is slightly smaller than for the equal model.

In the distances calculated based on the CV values displayed in Figure 4.3b we see a similar trend. As resolution increases the distances between the discrete models and the first seeded simulation of the continuous model become smaller. The droop and the equal model achieve the smallest distance, 0.039 and 0.034, for resolution $\frac{1}{128}$. With the distance for the equal model falling within the range of the reference distances $0.027 \pm 0.011$. Which of the equal and the droop model has the smallest distance changes for different resolutions. However it seems like the droop model has the smallest distance for all resolutions except for resolution $\frac{1}{128}$. However, it is worth mentioning that the difference between the discrete models distances are less than 0.005 while the standard deviations of the distances are about 0.02.

The distances calculated based on the CC values displayed in Figure 4.3c, shows the droop model having a smaller distance than the equal model for all resolutions except the highest one. However, the difference between the distances of the discrete models are about 0.02 and the standard deviation of the distances are about 0.03. Both the droop and the equal model gets distances smaller than the reference values as we reach resolutions $\frac{1}{8}$ and $\frac{1}{16}$, respectively. They achieve the smallest distances for resolution $\frac{1}{16}$ for the equal model and $\frac{1}{32}$ for the droop model. However as the resolutions increase further we can see that the distances get larger again.

### 4.1.2 The effect of discretizing delay values

In this section we explore a version of the Brunel network where the spike times are continuous, but the delay values are discrete, drawn from the interval [min, max]. We display the mean distributions of the statistics as ecdf curves, and as functions of resolution in a similar matter to what we did in the Brunel network.



**(a)** Wasserstein distance based on FR values.

**(b)** Wasserstein distance based on CV values.

**(c)** Wasserstein distance based on CC values.

**Figure 4.3:** The Wasserstein distances between the discrete models and the first seeded simulation of the continuous model are displayed in purple for the droop model and dashed blue lines for the equal model. The distance between the first seeded simulation of the continuous model and all other seeded simulations in that model is displayed by the green line. The distances have been calculated based on three statistics: Firing Rate, Coefficient of Variation, and the Correlation Coefficient.

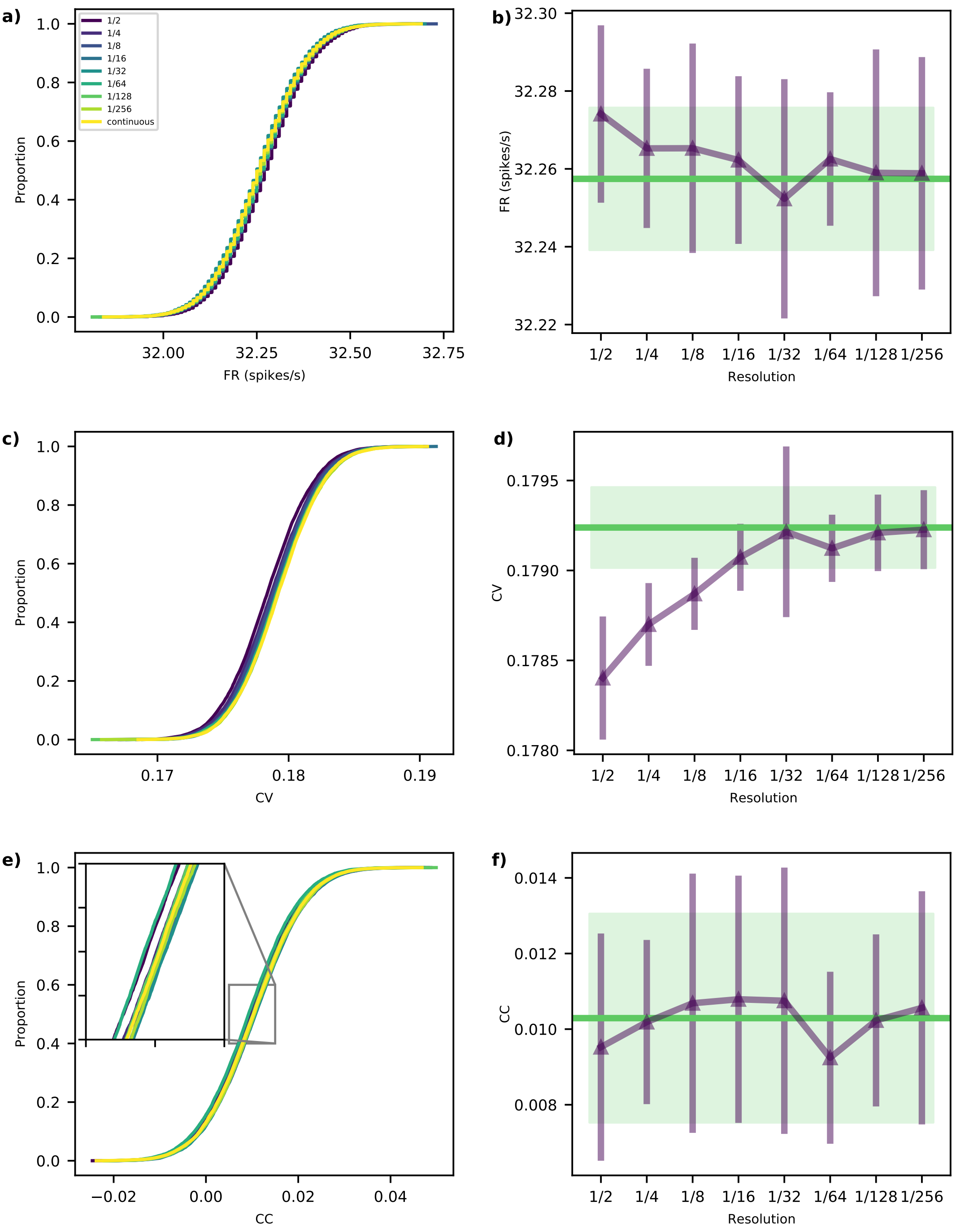


**Figure 4.4:** From the top: mean FR, CV and CC distributions for the semi-continuous and the continuous model. Both as ecdf curves and as functions of resolution. The ecdf curves on the left shows the semi-continuous model for each resolution, with the colors represented in the legend of the top left figure. The yellow curve represents the continuous case. On the right hand side, the semi-continuous model is represented in purple, and the continuous model is in green. The vertical lines represents the variation in the values and the shaded green area shows the variation of the continuous model's mean value.

If we examine the ecdf curves in Figure 4.4 on the left hand side, we can see that there are several ecdf curves, which are not identical. They do however lay close together. There is in total nine ecdf curves, one for each resolution of the semi-continuous model, and one for the continuous model. The color representing each resolution is displayed in the legend in the top left figure. The continuous model is represented in yellow. For the mean FR distributions in the top left figure, one can see that the curves are spread around the curve of the continuous model on both sides. However, they are closely gathered. One can not tell which curves lie the closest to the one representing the continuous model, as most of them are overlapping.

Also in the case of the mean CV distributions, in the middle left plot of Figure 4.4, the ecdf curves are not that easily distinguishable. It does however look like the curve for the lowest resolution is the furthest away from the curve of the continuous model. Then the curves representing higher resolutions lie closer to the curve of the continuous model. For the CC distributions in the bottom left of Figure 4.4 the curves are even harder to distinguish, they are covering each other. But it seems like the curve representing the highest resolution is overlapping the most with the curve of the continuous model. One can see this in the window displaying a zoomed in version of the curves.

We try to get a better overview by examining the mean values of the statistics as functions of resolutions. They are displayed in the figures on the right hand side of Figure 4.4. For the FR values, displayed in the top right plot, the two models seem to have the same means for resolutions $\frac{1}{128}$ and $\frac{1}{256}$, they have a difference of 0.0015 spikes/s. The difference between the two models means are always less than 0.017 spikes/s, while the standard deviation is of 0.025 spikes/s. That is, for all resolutions, the mean FR of the semi-continuous model is within the variance range of the mean FR value of the continuous model. The mean of the semi-continuous model lies within the green shaded area.

When examining the mean CV values in the middle right plot of Figure 4.4, we can see that the semi continuous and the continuous model have quite different means for the lowest resolutions. The difference between the means is about 0.0008 for the lowest resolutions, but as we get to resolution $\frac{1}{32}$ they seem equal. Then the difference between the means is 0.00002. They are also very similar for the two highest resolutions, with a difference of 0.00003 and 0.00001. Note that the standard deviation is of size 0.0004. For resolutions above $\frac{1}{8}$, the mean CV value of the semi-continuous model falls within the variance range of the mean CV value of the continuous model. For the CC values in the bottom right plot of Figure 4.4, the two models seem to have the same mean for resolutions $\frac{1}{4}$ and $\frac{1}{128}$, the differences from the continuous model's mean are 0.0001 and 0.00006, respectively. In this case the means of the semi continuous model seem to lay close to the mean of the continuous one for all resolutions. The difference between the mean CC values is always less than 0.001 while the standard deviation is of size 0.003. The mean CC value of the semi-continuous model always fall within the variance range of the mean CC value of the continuous model.

We give the values of the mean FR, CV and CC values of the continuous and semi-continuous model for all resolutions in Table 4.2.

## 4.2 Microcircuit Network

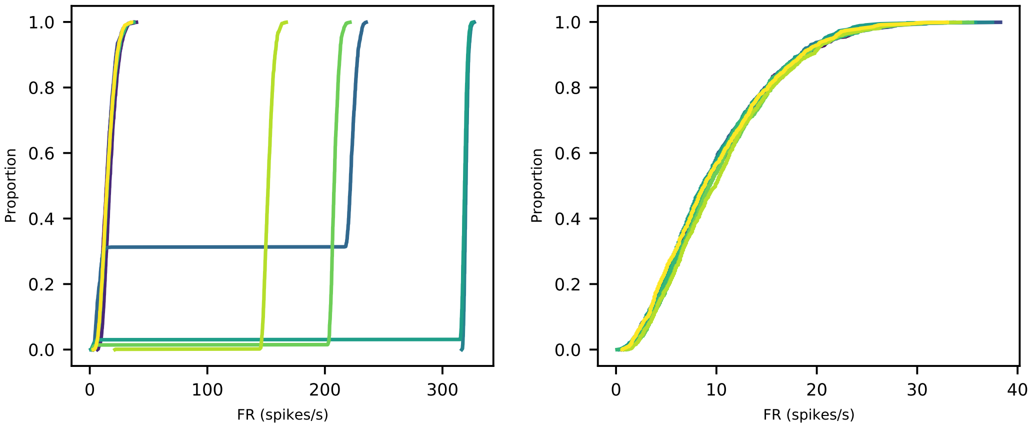
In contrast to the Brunel network model, the results of the Microcircuit network model will only be presented for resolutions $\frac{1}{8}, \ldots, \frac{1}{256}$. This is because for resolution $\frac{1}{2}$, the probability of each delay value is already equal when drawing from the interval [min, max]. To give more details: for excitatory delays we draw from the interval $[0.75, 2.25]$ for the droop model, and when the resolution is $\frac{1}{2}$ NEST creates delays with the possible values $1, 1.5, 2$. When drawing from the interval $[0.75, 2.25]$ the two endpoints 1 and 2 already have the same probability of being drawn as the middle point has, since $1 - h/2 = 0.75$ and $2+h/2 = 2.25$. The inhibitory delays are drawn from the interval $[0.375, 1.125]$, but for resolution $\frac{1}{2}$ the possible delay values become $0.5, 1$. Hence the probability of drawing the endpoint values are equal as we are drawing from the interval $[0.375 = 0.5 - h/2, 1.125 = 1 + h/2]$.

In the case of resolution $\frac{1}{4}$, we choose to leave this out as we observe extreme behaviour. The firing rates of both the equal and the droop model are extremely high for resolution $\frac{1}{4}$, and then becomes lower and more stable for the higher resolutions. Therefore, to get results that are comparable, we only include higher resolutions. We illustrate the extreme firing rates found for resolution $\frac{1}{4}$ in the droop model and compare it to the firing rates of the droop model when resolution is $\frac{1}{8}$ in Figure 4.5. In these figures the ecdf curve of the FR values of every seeded simulation in the model is included. We can see that the ecdf curves of the firing rates for the model with resolution $\frac{1}{8}$, in Figure 4.5b, all lie close together ranging from 0 to 40 spikes/s. While for resolution $\frac{1}{4}$, in Figure 4.5a, the curves are more spread out, differing a lot for different choices of the seed value, with some of them having firing rates up to 300 spikes/s.

In a similar matter to the presentation of the results of the Brunel network, we visualize the mean statistics of the three different models together in plots. We include one statistic in each figure. Both the ecdf curves and the mean statistics as a function of resolution are visualized in Figure 4.6. The same colors and line styles are used here to distinguish between the different models and resolutions as were used in the case of the Brunel network model. That is, in the figures on the left hand side, the ecdf curves, the equal model is represented with dashed lines and the droop model is represented with solid lines. The equal and the droop model have the same color for the same resolution.

| | | Resolutions | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | 1/2 | 1/4 | 1/8 | 1/16 | 1/32 | 1/64 | 1/128 | 1/256 |
| Semi-Continuous | FR (spikes/s) | 32.274 | 32.265 | 32.265 | 32.262 | 32.252 | 32.263 | 32.259 | 32.259 |
| | CV | 0.1784 | 0.1787 | 0.1789 | 0.1791 | 0.1792 | 0.1791 | 0.1792 | 0.1792 |
| | CC | 0.0095 | 0.0102 | 0.0107 | 0.0108 | 0.0108 | 0.0092 | 0.0102 | 0.0106 |
| Continuous | FR (spikes/s) | - | - | 32.257 | - | - | - | - | - |
| | CV | - | - | 0.1792 | - | - | - | - | - |
| | CC | - | - | 0.0103 | - | - | - | - | - |

**Table 4.2:** The mean FR, CV and CC value of the semi-continuous and continuous model for all resolutions in the Brunel network.



**(a)** Resolution $\frac{1}{4}$

**(b)** Resolution $\frac{1}{8}$

**Figure 4.5:** The ecdf curves of the firing rates for each seeded simulation in the Microcircuit network, droop model.

The continuous model is displayed in yellow. One can see which colors represent which resolutions in the legend in the top left figure. On the right hand side the mean statistics and variance as a function of resolution are displayed. The purple curve represent the droop model, the dashed blue curve represent the equal model, and the green line shows the mean statistic of the continuous model. The variance of the means are displayed by vertical bars in the same color as the mean values, with the continuous model's variance displayed by the shaded green area.

When examining the FR distributions in the top left plot of Figure 4.6, we observe that the curve for the droop model simulated with resolution $\frac{1}{8}$, lies further away from the other curves. It has a difference of around 1.14 spikes/s from the main cluster of curves. The curve for the droop model with resolution $\frac{1}{16}$ also has some distance from the main cluster of curves, with a difference of around 0.5 spikes/s. The rest however, lie closer together, differing by maximum 0.34 spikes/s and are not that easy to distinguish.

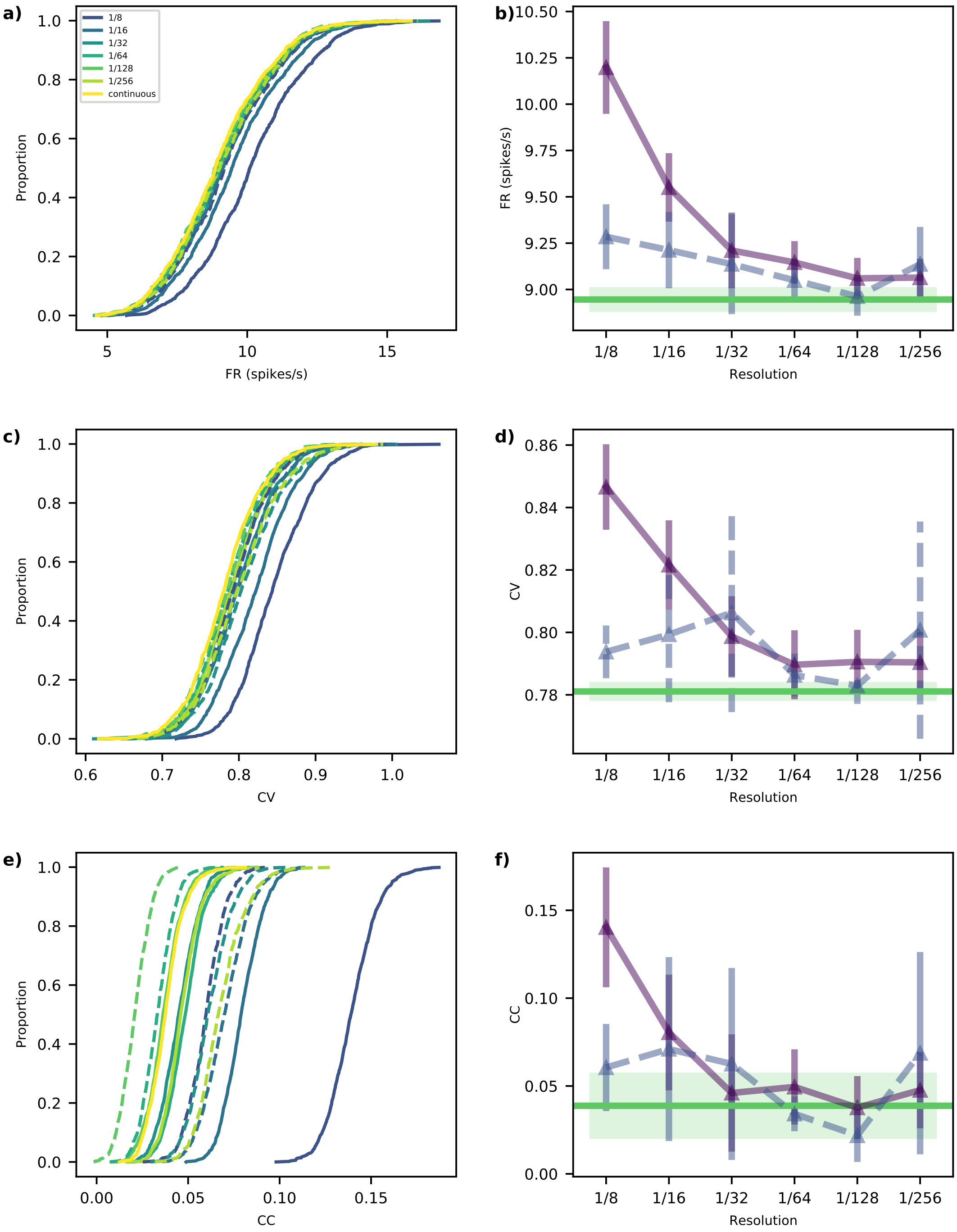


**Figure 4.6:** From the top: mean FR, CV and CC distributions for all three cases, droop, equal and continuous. Both as ecdf curves and as functions of resolution. The ecdf curves on the left shows the equal model in dashed lines, the equal and the droop model have the same color for the same resolutions. The yellow curve represents the continuous case. On the right hand side, the droop model is represented as purple, the equal model as dashed blue lines, and the continuous model in green. The vertical lines represents the variation in the values and the shaded green area shows the variation of the continuous model's mean value.

We observe a similar trend in the ecdf curves of the mean CV distribution for all models in the middle left plot of Figure 4.6. We can see two curves being further away from the main cluster of curves. These curves represent the droop model for resolutions $\frac{1}{8}$ and $\frac{1}{16}$, having a difference of around 0.05 and 0.03 from the main cluster of curves. The other curves are more gathered with a maximum difference of around 0.018. In the case of the mean CC distributions however, displayed in the bottom left plot of Figure 4.6, one can more easily distinguish the different curves from each other. Furthest away from the continuous model displayed in yellow, we find the droop model for resolution $\frac{1}{8}$ (displayed in dark blue). There is a difference of around 0.1. For resolution $\frac{1}{16}$ the mean CC curve of the droop model come closer to the continuous one, having a difference of around 0.04. For the higher resolutions, the curves lie closer together, even closer to the continuous model. The curve for the droop model with resolution $\frac{1}{128}$ lies the closest to the curve of the continuous model compared to the other curves. Having a mean difference of 0.001, they are almost overlapping. For the equal model, the ecdf curves of the CC distributions are spread on both sides of the continuous model. Among these, the curve for the equal model with resolution $\frac{1}{64}$ is the closest to the continuous model's curve with a difference of 0.005.

We get a more clear overview of the differences between the droop and the equal model from the continuous model in terms of the statistics $FR$, $CV$ and $CC$ when examining the mean statistics as functions of resolution. They are displayed in the plots on the right hand side of Figure 4.6. The mean FR values are displayed in the top right figure. One can see a stable trend in the difference between the discrete models and the continuous one. The line representing the equal model's mean FR value lies closer to the line of the continuous model than the droop model does, for all resolutions except the highest one. The equal model has a difference from the continuous model's mean FR value of less than 0.4 spikes/s for all resolutions, and achieves the smallest difference for resolution $\frac{1}{128}$ with a difference of 0.017 spikes/s. The droop model has a difference of around 1.25 spikes/s from the continuous model's mean FR value for the lowest resolution, but decreases to less than 0.27 spikes/s as the resolution becomes $\frac{1}{32}$. It achieves the least difference from the continuous model's mean FR value for resolution $\frac{1}{128}$ with a difference of around 0.11 spikes/s. The difference in mean FR values between the droop and the equal model are around 0.9 spikes/s for the lowest resolution, then this difference decreases as resolution increases until the highest one where they achieve similar means with a difference of 0.07 spikes/s between them. The standard deviations of the means of the discrete models are approximately 0.25 spikes/s, while the standard deviation of the continuous model's mean FR value is about 0.06 spikes/s.

The mean CV values are displayed in the middle right plot of Figure 4.6. For the lowest resolution, one can observe that the curve representing the droop model's mean CV is the furthest away from the continuous model's mean. It has a difference of 0.065, while the equal model has a difference of 0.013 from the continuous model's mean. The curve representing the mean CV value of the equal model is closer to the continuous model's mean for the two lowest resolutions, but then the line representing the droop model gets closer for resolution $\frac{1}{32}$ having a difference of 0.018, while the equal model has a difference of 0.025 from the continuous model's mean CV. As we get to resolution $\frac{1}{64}$, the curve of the equal model gets closer to the continuous model's mean, with differences less than 0.01. Then for the highest resolution, the curve of the droop model's mean CV is closer to the continuous one with a difference of 0.009, while the equal model's mean CV has a difference of 0.02 from the continuous model's mean.

We can examine the mean CC values in the bottom right plot of Figure 4.6. The line representing the droop model is the furthest away from the continuous model's mean CC value for the two lowest resolutions, with a difference of 0.1 and 0.04. It then becomes closer to to the continuous model's mean CC value, out of the two discrete models, for the higher resolutions with a difference of less than 0.01. For resolution $\frac{1}{64}$ however, the equal model's mean has the least difference from the continuous model's mean, with a difference of 0.005. This is the resolution for which the equal model has the least difference in means from the continuous model. The droop model has its mean closest to the continuous model's mean for resolution $\frac{1}{128}$ with a difference of 0.001. The equal models mean CC value is never more different than about 0.03 from the continuous model's mean. Note that the standard deviations of the discrete models ranges a size of between 0.01 and 0.06.

The values of the mean statistics of the droop, equal and continuous model for each resolution, are included in Table 4.3.

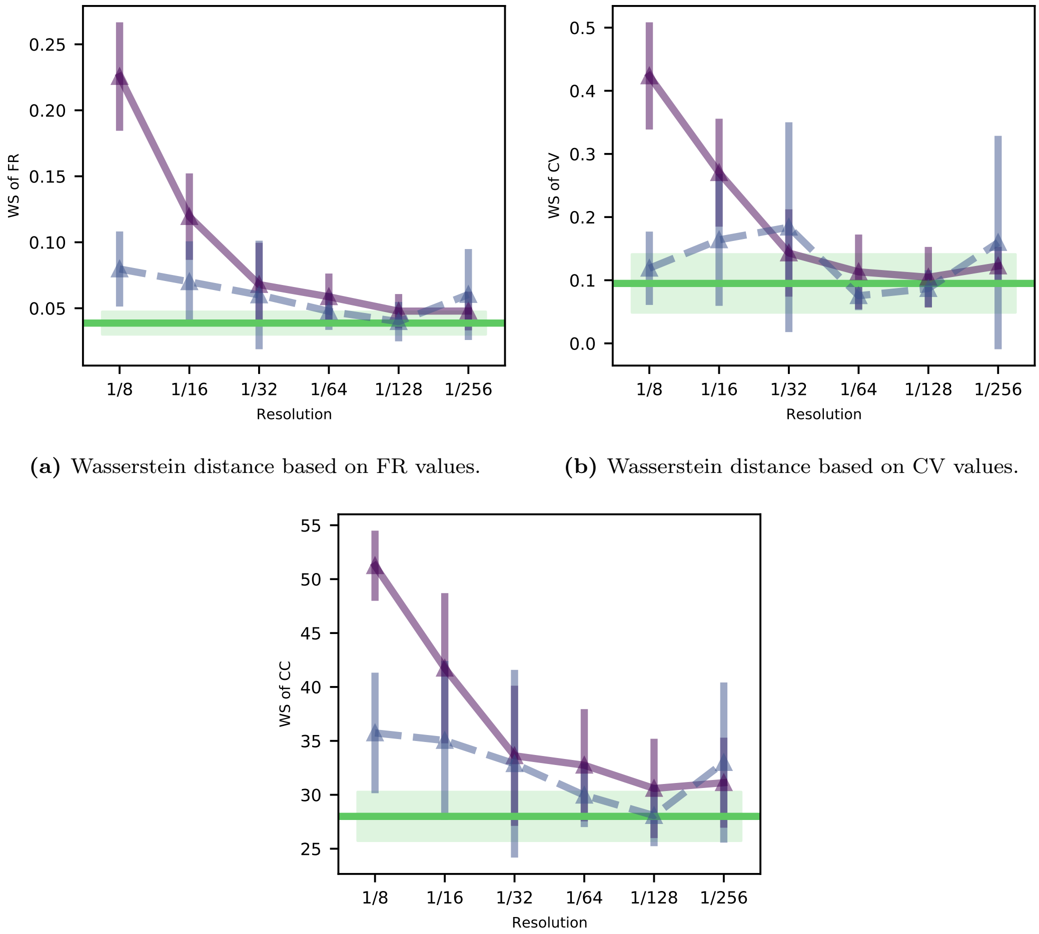
### 4.2.1 Wasserstein Distances

To make the differences between the three models more clear, the Wasserstein distance metrics have been calculated. The average of these distances and the standard deviation of them was taken and are displayed in the figures seen in Figure 4.7.

If we examine the plots of Figure 4.7, we can see three lines. The purple line represents the distance between the droop model and the first seeded simulation of the continuous model, the dashed blue line represents the distance between the equal model and the first seeded simulation of the continuous model, and the green line represent the distance between the continuous model all seeded simulation vs. the first seeded simulation. The values of the lines is the mean Wasserstein distances. The variation in the distances of the continuous simulations are displayed by the shaded green area, while the variation of the distances of each of the droop and equal model versus the first seeded sim-

| | | Resolutions | | | | | |
|---|---|---|---|---|---|---|---|
| | | 1/8 | 1/16 | 1/32 | 1/64 | 1/128 | 1/256 |
| Droop | FR (spikes/s) | 10.198 | 9.551 | 9.211 | 9.146 | 9.060 | 9.064 |
| | CV | 0.8466 | 0.8216 | 0.7987 | 0.7896 | 0.7906 | 0.7904 |
| | CC | 0.1403 | 0.0805 | 0.0460 | 0.0495 | 0.0377 | 0.0476 |
| Equal | FR (spikes/s) | 9.285 | 9.213 | 9.136 | 9.050 | 8.963 | 9.136 |
| | CV | 0.7938 | 0.7993 | 0.8063 | 0.7863 | 0.7829 | 0.8007 |
| | CC | 0.0605 | 0.0710 | 0.0625 | 0.0341 | 0.0218 | 0.0687 |
| Continuous | FR (spikes/s) | 8.946 | - | - | - | - | - |
| | CV | 0.7811 | - | - | - | - | - |
| | CC | 0.0387 | - | - | - | - | - |

**Table 4.3:** The mean FR, CV and CC value of the droop, equal and continuous model for all resolutions in the Microcircuit network.



**(a)** Wasserstein distance based on FR values.

**(b)** Wasserstein distance based on CV values.

**(c)** Wasserstein distance based on CC values.

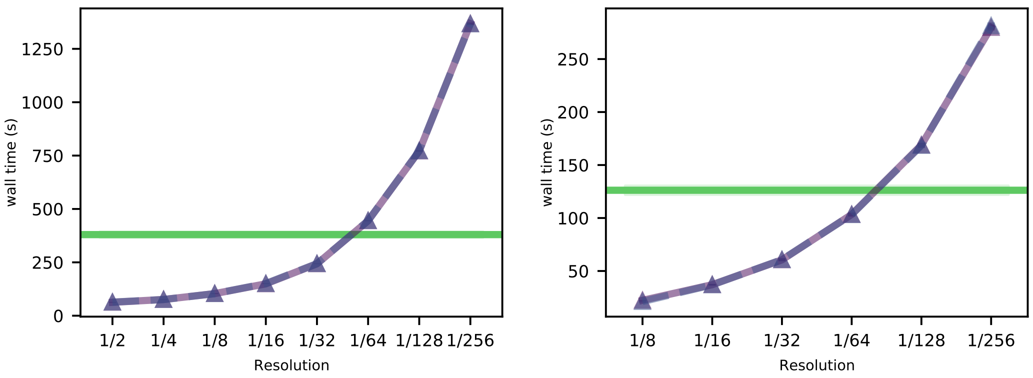
**Figure 4.7:** The Wasserstein distances between the discrete models and the first seeded simulation of the continuous model are displayed in purple for the droop model and dashed blue lines for the equal model. The distance between the first seeded simulation of the continuous model and all other seeded simulations in that model is displayed by the green line. The distances have been calculated based on three statistics: Firing Rate, Coefficient of Variation, and the Correlation Coefficient.

ulation of the continuous model are displayed by vertical lines. The mean and variance of the continuous models distances can be used as a reference point of what distances to expect when two simulations are of the same model. Hence giving us an idea of what is viewed as a small distance.

When the Wasserstein distances are calculated based on the firing rates, displayed in Figure 4.7a, the reference distances are $0.04 \pm 0.01$. Furthermore, the equal model seems to have a smaller distance than the droop model for all resolutions, except for the highest one. The equal model's distances are higher than the reference distances for most resolutions, but get close for resolutions $\frac{1}{64}$ and $\frac{1}{128}$, where the distances are 0.048 and 0.04. That is, the distance between the equal model and the first seeded simulation of the continuous model, falls within the range of the reference distances for resolution $\frac{1}{64}$ and $\frac{1}{128}$. The distances for the droop model are at their closest to the mean reference distance for resolutions $\frac{1}{128}$ and $\frac{1}{256}$, having distances of approximately 0.048. That is, the droop model's distances are falling within the range of reference distances. It has the least distance of the two discrete models for the highest resolution.

When examining the Wasserstein distances based on the CV values of the models in Figure 4.7b, we can see that the mean distance for the continuous model against itself, is $0.095 \pm 0.046$. The distance between the droop model and the first seeded simulation of the continuous model, is quite high for the two lowest resolutions, with distances of approximately 0.4 and 0.3. The equal model however, starts out with a small distance of about 0.12, that is, the distance lies within the area of the reference distances. Then it increases. For resolutions $\frac{1}{64}$ and $\frac{1}{128}$ both of the discrete models have distances within the range of reference distances. They are approximately 0.11 and 0.1 for the droop model, and 0.075 and 0.087 for the equal model. Then the distance for the equal model becomes larger for the highest resolution with a distance of 0.16, while the droop model's distance is 0.12, still lying within the reference area.

In the case of the Wasserstein distances based on the correlation coefficients, displayed in Figure 4.7c, the mean reference distance is about $28 \pm 2.27$. The distances of the equal model is smaller than for the droop model for all resolutions except the highest one. The distances between the droop model and the first seeded simulation of the continuous model for the lower resolutions are between 51 and 32. Only when we reach resolutions $\frac{1}{128}$ and $\frac{1}{256}$ does the distances get close to the area of reference values, the green shaded area, having distances of 30.59 and 31.12. For the lower resolutions the equal model has distances between 36 and 32, then they get inside the area of reference distances for resolutions $\frac{1}{64}$ and $\frac{1}{128}$, with a distance of 29 and 28. Note that for resolutions $\frac{1}{32}$ and above, the difference between the distances of the droop and the equal model is under 3 while the standard deviation of the distances are about 5.



**(a)** Wall time Brunel network.

**(b)** Wall time Microcircuit network.

**Figure 4.8:** The wall time for the two network models. They are displayed as the mean of the wall times as a function of resolution with the variance displayed with vertical lines and shaded area. The droop model is represented in purple, the equal model as a dashed blue line, and the continuous model in green.

## 4.3 Runtime

It is of interest to examine the run-time of the simulations of the network models in order to see how much more computational power is needed as resolution increases. The average wall times are displayed in Figure 4.8. The droop model is displayed as a purple line, the equal model as a dashed blue line, and the continuous model in green. The variance in the times are also displayed with vertical bars, although the variance is so small it can almost not be seen in this case. The figure on the left hand side displays the times for the Brunel network. We can see that the run times of the droop and the equal model are almost the same as their lines are overlapping, they do not differ by more than 1s. As the resolution reaches $\frac{1}{64}$, the run time of the two discrete models surpasses the run time of the continuous model. One does no longer save time by using discrete spike times and delay values when the resolution is this fine.

The figure on the right hand side displays the run times for the Microcircuit network model. In the case of the Microcircuit model one can see a little variation in the run times of the continuous model with a variation of 4s. The droop and the equal model has a variation in time of approximately 2s. One can see that the droop and the equal model have almost the same run times as their lines are overlapping, they never differ by more than 1s. But in this case the run times of the discrete models surpasses that of the continuous model as the resolution reaches $\frac{1}{128}$.

Note that the wall time of the microcircuit network model in Figure 4.8b, is much shorter than the wall time of the Brunel network model. This might be contradictory to what one expects as the microcircuit model has over 77000 neurons while the Brunel model only has 10000 neurons. The simulations were however run differently, as the Brunel model was ran with 1 threads per task, while the Microcircuit network was ran with 32 threads per task.

# Chapter 5

# Discussion

When the delay values are drawn from discrete intervals, is it important that every single value has the same probability of being drawn? Or does it not matter if the maximum and minimum delay value only have half the probability of being drawn? Will this difference in probabilities affect the spiking activity to such a degree that one can say one version is more correct in regards to our reference spiking activity, where the delay values are drawn from a continuous interval? And when this is answered, is there a choice of resolution which ensures that the spiking activity of a discrete model is similar to that of a continuous model, while still being computationally cheaper? We try to answer this by going through our findings.

## 5.1 Key findings

In Sec. 4.1 we explored the mean distributions of the statistics, FR, CV and CC of the Brunel network. We compared the two rounding rules to the continuous model for several resolutions. We saw that not one of the discrete models were closest to that of the continuous model based on the statistics mentioned. At least not for all resolutions. That is, whether the droop or the equal model had statistics the closest to the continuous model changed for resolutions and the choice of statistic. Although for the FR and CV values it seemed like the droop model was closer to the continuous model for most resolutions, this was not the case for all resolutions. And the differences between the means of the droop and the equal model were not that large when taking the size of the variances into account. We also saw that as we reached resolution $\frac{1}{64}$, the statistics of the droop and the equal model got quite close to that of the continuous model. They got within the area of the continuous model's mean for resolution $\frac{1}{128}$ for the CV values, and $\frac{1}{256}$ for the FR values. For the CC values the discrete models means got within the continuous model's mean area for resolutions $\frac{1}{4}$ and up.

We then went on to explore the Wasserstein distances between the discrete models and the continuous model in Sec. 4.1.1. They were calculated based on the statistics we explored in the preceding section. The droop model seemed to have a smaller distance to the continuous model than the equal model did, for all resolutions except one. The difference in the distances of the discrete models did however not seem to be that large when taking the size of the variances into account. In the case of the distances calculated based on the FR and CV values, the distance of the discrete models to the continuous model got small for resolution $\frac{1}{64}$, being quite close to the reference distances. However, they actually reached the reference distances as we get to resolution $\frac{1}{256}$ for the FR values and $\frac{1}{128}$ for the CV values. While for the distance calculated based on the CC values, the models reached the reference distances for resolution $\frac{1}{8}$ and up.

In Sec. 4.1.2 we explored the effects of making only the delay values discrete, and not the spike times in the Brunel network. There were differences between the mean statistics, these differences were however not that large when taking the size of the variance into account. The mean of the semi-continuous model fell within the range of the mean of the continuous model. Only for the CV values was there a noticeable difference between the two models means for lower resolutions. However, they did fall within the same region as we reached the higher resolutions. Hence one could say there is no significant difference in the models activity in regard to their mean statistics as resolution becomes high enough, being $\frac{1}{32}$ and up. However for lower resolutions, differences can be seen. At least in the case of the CV values.

We then went on to explore the difference between the mean statistics of the droop, the equal and the continuous model in the Microcircuit network in Sec. 4.2. In this network, the equal model had mean statistics closest to the continuous model's mean for most resolutions in the case of the FR values. The mean got within the area of the continuous model's mean as we had resolution $\frac{1}{128}$, but got very close for resolution $\frac{1}{64}$. The droop model got closest to the reference area for resolution $\frac{1}{256}$ in the case of the FR values. For the CV and CC values however, the model having the mean closest to that of the continuous model changed a lot as resolution changed. The droop model's mean get close to the continuous model's mean as we reach resolution $\frac{1}{64}$ for the CV values, and gets within the continuous models mean area for the CC values for resolution $\frac{1}{32}$ and up. The equal model's mean falls within the area of the continuous model's mean for resolution $\frac{1}{128}$ in the case of the CV values, but gets very close for resolution $\frac{1}{64}$. For the CC values the equal model gets within the continuous model's mean area for resolution $\frac{1}{64}$.

In Sec. 4.2.1, we explored the differences in the statistics of the Microcircuit network by examining their Wasserstein distances. We saw that distances calculated based on the FR values, showed that the equal model was closer to that of the continuous one for all resolutions except one, reaching the reference distances for resolution $\frac{1}{64}$. The droop model's mean is very close to the reference distances for this resolution, however it does not reach the reference distances before resolution $\frac{1}{256}$. For the distances calculated based on the CV values, the model being the closest changed as resolutions changed. With the droop and the equal models distances from the continuous model falling within the reference area for resolution $\frac{1}{64}$. While for the CC values the equal model seemed to have smaller distances for all resolutions except one, falling within the reference area for resolution $\frac{1}{64}$. The droop model's distances never falls within the reference area, but got close for resolution $\frac{1}{128}$.

Then lastly we displayed the run times of the two networks explored in this thesis, the Brunel and the Microcircuit network, in Sec. 4.3. We saw that the two discrete models had the same run time for all resolutions, and that they surpassed the run time of the continuous model for resolution $\frac{1}{64}$ for the Brunel network, and $\frac{1}{128}$ for the Microcircuit network.

## 5.2 Limitations of our work

Due to limitations in time, several things were not explored. We quickly mention them here. In the case of the Wasserstein distances, we only found the distances between the simulations of the discrete models against the first seeded simulation of the continuous model. It could be of interest to get the distance to all seeded simulation of the continuous model. This way we would get measures of how distant the discrete model's are from the continuous model, and not just one simulation of the continuous model. In some cases the distances of the discrete models were very close to the reference distances, barely falling outside the reference area. Maybe they would fall within the area had we found the distance to all simulations of the continuous model. It should also be mentioned that we have only looked into a few statistics of the spiking activity, being the FR, CV and CC statistics. Maybe other aspects could have been seen if we had looked into other statistics. In addition to only looking into a few statistics, we also only examined a few neuronal populations. E.g. for the Brunel network we only looked into the excitatory population, and for the Microcircuit network, we only looked into the L5I population. Maybe we would we see something else if we had explored all neuronal populations of our network models. Lastly, it should be mentioned that we did explore the statistics of a semi-continuous version of the Brunel network, it could however be interesting to explore a semi-continuous version of the Microcircuit network as well. This was not done due to limitation in time and computation hours.

## 5.3 Conclusion

To conclude with one of the rounding rules being more correct in the sense that the spiking activity is more similar to that of the continuous model, one would have to see that one of the rounding rules mean statistics was closer to that of the continuous one for most if not all resolutions, for all statistics and both network models. But in our case we saw that the rounding rule being the closest to that of the continuous one was not the same for all resolutions. Even when one model was closer for most resolutions, this was not the case for all statistics. We also noticed that although it might look like one model was closer to the continuous one for several resolutions and statistics, this model was different for the two networks. E.g. it looked like the droop model was slightly closer to the continuous one for the FR and CV values in the Brunel network. However, for the Microcircuit network, the equal model seemed to be closer to the continuous one for the FR values for most resolutions. Hence the model that might seem to be closer was different for the two network models explored. One can not conclude with one rounding rule being more correct across all networks. Whether the minimum and maximum value of the discrete delay values interval has the same probability of being drawn as the inner points or not, is not critical for the spiking activity. It is not more correct to ensure equal probability of all delay values being drawn.

Furthermore, to save computational power by using a discrete model instead of a continuous one, the resolution should not be higher than $\frac{1}{32}$ for the Brunel network, and $\frac{1}{64}$ for the Microcircuit network. For the Brunel network, the statistics of the discrete models got close to that of the continuous model as resolutions were $\frac{1}{64}$ and up for the FR and CV values, while for the CC values the means are close for resolution $\frac{1}{32}$. For the Microcircuit network the statistics of the discrete models get quite close to that of the continuous one as we reach resolutions $\frac{1}{64}$. One could therefore achieve similar spiking activity for the discrete models as can be seen in the continuous model when using resolution $\frac{1}{64}$ for both the Brunel and the Microcircuit network models. This does however not save computation time for the Brunel network, as the discrete models with this resolution takes longer to simulate than the continuous model.

Through the work of this thesis we have explored the statistics of three model types in two different networks. We have seen that no clear differences can be seen between the two rounding rules of the discrete models. We have also explored how the activity changes and gets closer to that of the exact model as resolution increases. By examining this we also were able to give a hint of what resolutions are fine enough to create representative spiking activity. This could then be taken into account together with the computational cost when deciding on the choice of resolution in future simulations.

# References

Adobe Systems Incorporated (1999). *Postscript Language Reference Manual, third ed.* Reading, MA: Addison-Wesley Publishing Company.

Albers, J., J. Pronold, A. Christopher Kurth, S. B. Vennemo, K. H. Mood, A. Patronis, D. Terhorst, J. Jordan, S. Kunkel, T. Tetzlaff, M. Diesmann, and J. Senk (2021, December). A modular workflow for performance benchmarking of neuronal network simulations.

Brunel, N. (2000). Dynamics of sparsely connected networks of excitatory and inhibitory spiking neurons. *Journal of Computational Neuroscience 8*, 183–208.

Conover, W. (1999). *Practical nonparametric statistics* (3rd ed.). New York: John Wiley & Sons Inc.

Dasbach, S. (2020, January). Effects of discretized synaptic weights on the dynamics of recurrent spiking neuronal networks. Master's thesis, RWTH Aachen University.

Deepu, R., S. Spreizer, G. Trensch, D. Terhorst, S. B. Vennemo, J. Mitchell, C. Linssen, H. Mørk, A. Morrison, J. M. Eppler, N. L. Kamiji, R. de Schepper, I. Kitayama, A. Kurth, A. Morales-Gregorio, P. Nagendra Babu, and H. E. Plesser (2021, September). Nest 3.1.

Denker, M., A. Yegenoglu, and S. Grün (2018). Collaborative HPC-enabled workflows on the HBP Collaboratory using the Elephant framework. In *Neuroinformatics 2018*, pp. P19.

Devore, J. L. and K. N. Berk (2012). *Regression and Correlation*, pp. 663. New York, NY: Springer New York.

Eppler, J., M. Helias, E. Muller, M. Diesmann, and M.-O. Gewaltig (2009). Pynest: a convenient interface to the nest simulator. *Frontiers in Neuroinformatics 2*.

Gabbiani, F. and C. Koch (1998). Principles of spike train analysis. In C. Koch and I. Segev (Eds.), *Methods in neuronal modeling : from ions to networks* (2nd ed.)., Computational neuroscience, Chapter 9, pp. 313–338. Cambridge, Mass.: MIT Press.

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

Gustavsson, A., M. Svensson, F. Jacobi, C. Allgulander, J. Alonso, E. Beghi, R. Dodel, M. Ekman, C. Faravelli, L. Fratiglioni, B. Gannon, D. H. Jones, P. Jennum, A. Jordanova, L. Jönsson, K. Karampampa, M. Knapp, G. Kobelt, T. Kurth, R. Lieb, M. Linde, C. Ljungcrantz, A. Maercker, B. Melin, M. Moscarelli, A. Musayev, F. Norwood, M. Preisig, M. Pugliatti, J. Rehm, L. Salvador-Carulla, B. Schlehofer, R. Simon, H.-C. Steinhausen, L. J. Stovner, J.-M. Vallat, P. V. den Bergh, J. van Os, P. Vos, W. Xu, H.-U. Wittchen, B. Jönsson, and J. Olesen (2011, oct). Cost of disorders of the brain in Europe 2010. *European Neuropsychopharmacology 21*(10), 718–779.

Gutzen, R., M. von Papen, G. Trensch, P. Quaglio, S. Grün, and M. Denker (2018). Reproducible neural network simulations: Statistical methods for model validation on the level of network activity data. *Frontiers in Neuroinformatics 12*.

Joyce, J. M. (2011). *Kullback-Leibler Divergence*, pp. 720–722. Berlin, Heidelberg: Springer Berlin Heidelberg.

Jülich Supercomputing Center (2021). Jülich supercomputing center.

Kanwal, R. P. (1983). Generalized functions : theory and technique.

Kriener, B., H. Enger, T. Tetzlaff, H. E. Plesser, M.-O. Gewaltig, and G. T. Einevoll (2014). Dynamics of self-sustained asynchronous-irregular activity in random networks of spiking neurons with strong synapses. *Front Comput Neurosci 8*, 136.

Morrison, A., S. Straube, H. E. Plesser, and M. Diesmann (2007). Exact subthreshold integration with continuous spike times in discrete time neural network simulations. *Neural Computation 19*, 47–79.

Nordlie, E., M.-O. Gewaltig, and H. Plesser (2009, 09). Towards Reproducible Descriptions of Neuronal Network Models. *PLoS computational biology 5*, e1000456.

Potjans, T. C. and M. Diesmann (2012, 12). The Cell-Type Specific Cortical Microcircuit: Relating Structure and Activity in a Full-Scale Spiking Network Model. *Cerebral Cortex 24*(3), 785–806.

Ramdas, A., N. G. Trillos, and M. Cuturi (2017). On wasserstein two-sample testing and related families of nonparametric tests. *Entropy 19*(2).

Sterratt, D., B. Graham, A. Gillies, and D. Willshaw (2011). *Principles of computational modelling in neuroscience* (1st ed.). Cambridge, UK: Cambridge University Press.

Van Rossum, G. and F. L. Drake (2009). *Python 3 Reference Manual*. Scotts Valley, CA: CreateSpace.

Wnuk, A., A. Davis, C. Parks, D. K. Deborah Halber, G. Zyla, K. Hopkin, K. Weintraub, J. M. Beverly, K. S. Sheikh, L. Wessel, L. Chiu, M. Fessenden, M. Galinato, M. Richardson, and S. R. Sandra Blumenrath (2018). *Brain Facts, A primer on the brain and nervous system* (8th ed.). Washington DC: The Society of Neuroscience.

# Acknowledgments

First, I want to thank Prof. Dr. Hans Ekkehard Plesser for all guidance and help during the work of this master thesis. I would also like to thank Stine Brekke Vennemo and Håkon Mørk for teaching me how to use the supercomputers and the beNNch framework. In addition, I would like to thank Robin Gutzen and Aitor Morales-Gregorio for the help with setting up and using the NetworkUnit package, and for the fast and helpful answers to all questions.

We acknowledge the use of Fenix Infrastructure resources, which are partially funded from the European Union's Horizon 2020 research and innovation programme through the ICEI project under the grant agreement No. 800858.

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

Postboks 5003
NO-1432 Ås
Norway