DONDERS INSTITUTE

Radboud Universiteit

Master Thesis

To spike or not to spike: from a change of the excitability type in single neurons to exploring coding properties in network models

Raimon Bullich Vilarrubias
Nijmegen, The Netherlands
August, 2022
s1065452

Supervisors:
Fleur Zeldenrust and Sander Keemink
Biophysics of Neural Computation and *Artificial Cognitive Systems* Labs

Second reader:
Bernhard Englitz
Computational Neuroscience Lab

Abstract

From single neurons to large neural networks, the brain is constantly carrying and transforming information. Neural coding is a core aspect of information representation, which relies on the stimulus-response relationship. However, the neural coding properties at a single neuron and network level are still not fully understood.

At a single neuron level, the distinction between type 1 (integrator) and type 2 (resonator) neuron excitability is well known and relates to what causes a neuron to spike. However, it is unclear whether the excitability type of a single neuron can shift from type 1 to type 2 due to Acetylcholine (ACh) neuromodulatory effects.

Pushing this further, understanding how such single neuron coding properties could arise and interact at a network level is a great challenge considering the available network recording techniques. Here, network modeling is crucial for understanding network behaviour in the brain. With such models, one can study larger populations of neurons and still not lose information about single neuron properties. For instance, with the Poisson Balanced Spiking network (*Poisson* BSN) model, one can reproduce bio-like network spiking behaviour and still be able to observe single neuron properties. Yet, how the coding properties of neurons from network models arise and are affected by network properties is still unknown.

To study the coding properties in single neurons, we used the spike-triggered average (STA) analysis. With this, we investigated the stimulus-response relationship from single neuron in vitro recordings and neurons from network model simulations. From the experimental data analysis, we found that ACh did not cause a shift from type 1 to type 2 excitability. And regarding the network modelling we found that in the *Poisson* BSN model, network size, spiking cost, and time delays had an effect on changing the neurons' stimulus-response relationship given by the STA analysis.

Our results will allow for future research to model networks with neurons with different coding properties (such as type 1 and type 2) and account for how the interactions between neurons might affect these coding properties. Thus, this is one more step to developing and understanding network models to make them more biologically plausible and more insightful for studying and reproducing how the brain represents information.

Contents

Introduction

The brain is a very complex system that constantly processes and represents information by carrying and transforming it throughout the nervous system [1]. Hence, how input and output relate at any brain scale, from single neurons to neural networks and brain regions, is a central question in studying information representation. Generally, we can distinguish two main approaches to investigating information representation. One consists of reconstructing the input given the output (decoding). Alternatively, one could do the opposite and determine what relevant features of the input caused the output, so how the input is encoded to generate the output (encoding) [2]. Although both approaches are useful in studying information representation in the brain, how information is encoded and decoded is still an open question. This thesis focuses on the relationship between encoding and decoding in the brain, at both single neuron and network levels.

The highly heterogeneous nature of the brain is sometimes overlooked when investigating the coding properties of neurons and networks. Brain regions with diverse functionalities, differences in the morphological structures of its neurons, or distinctive excitability types between neurons are a few examples of heterogeneity in the brain [3]. It appears that neural heterogeneity is central to efficient and robust information representation rather than being the product of randomness or noisy processes [4, 5, 6]. Therefore, to understand how the brain carries and transforms information at a single neuron and network level, heterogeneity should not be ignored.

One of the most prominent examples of heterogeneity regarding neural coding is the distinction between type 1 (integrators) and type 2 (resonators) neurons [7, 8]. Type 1 neurons are sensitive to any positive input, and type 2 neurons are sensitive to the input's frequency. This distinction is invaluable in describing spike generation mechanisms in neurons and how the dynamics of neural excitability work (for a thorough discussion, see [9]). This difference in neural spike generation mechanisms is determined by different biophysical properties of the neurons [10]. Therefore, these properties play a very important role in how neurons can respond differently to a given input, segregating thus the coding properties of the neurons.

In in vitro experiments and neuron models, it has been found that by changing the neuron's biophysical properties, they can shift from type 1 to type 2 excitability, namely by modulating the voltage potassium currents (I_M) [11]. The biophysical properties of neurons are known to be affected by neuromodulators in the brain (for a review, see [12]). However, little attention has been paid to whether this shift can be mediated by them. Acetylcholine (ACh) is a neuromodulator that binds to the neuron's M1-receptors, which are in charge of regulating the flow of the I_M currents. Therefore, the first question of this thesis is: can ACh mediate a shift from type 1 to type 2 excitability in single neuron in in vitro recordings? (See orange stream of figure 1).

Understanding information representation in the brain should not be limited to single neurons but also aim at explaining it at a network level. On the one hand, investigating single neurons provides important insights into the mechanistic and dynamical processes that govern information

representation in single neurons. Investigating larger neural networks in the brain, on the other hand, can bring further knowledge about neural computations that cannot be observed in single neurons. Yet, there is still a gap between the mechanistic study of single neurons and the rather observational nature in the study of large neural networks [13]. A well-established approach to bridge this gap is the theoretical modeling of neural networks, which aims at describing how computations could be implemented in the brain in theoretically derived models of neurons (e.g., [14]). However, no focus has been put on how the coding properties of the neurons forming these network models arise and interact due to network effects. The Poisson Balanced Spiking Network framework (*Poisson* BSN) [15] might be a powerful framework to study neural coding in a network context for three main reasons. First, this model can be very useful to unravel how neurons communicate to perform computations, and the network dynamics underlying these computations are well understood. Second, the model captures random-like spiking activity seen in the brain while having efficient spiking and being robust to perturbations. And third, the network parameters of the Poisson BSN, namely network size, spiking cost terms, and connectivity delays might affect the firing properties of the neurons. Therefore, these parameters can be responsible for the changes in the neurons' coding properties. With this, the second question of this thesis is: how does the stimulus-response relationship of the *Poisson* BSN neurons arise and interact by tuning the network size, spiking cost, and connectivity delays? (see green stream of figure 1).

The coding properties of single neurons and how they are influenced at a network level are still open questions. In this research, we will approach these questions from two different angles, each one corresponding to a chapter of this thesis. One approach is analyzing single neuron data from neurophysiological in vitro recordings using spike-triggered averages (STA). With this, we will test whether the shift from type 1 to type 2 can be mediated by ACh. The other approach is network modelling of the *Poisson* BSN framework to understand how coding properties arise and interact at a network level. In the second approach, the same analysis (STA) will be applied. Both chapters contain four sections: background, methods, results, and discussion; and, at the end of the thesis, a conclusion will wrap up the results and discussion from both chapters, aiming at building bridges between them and proposing future research directions.

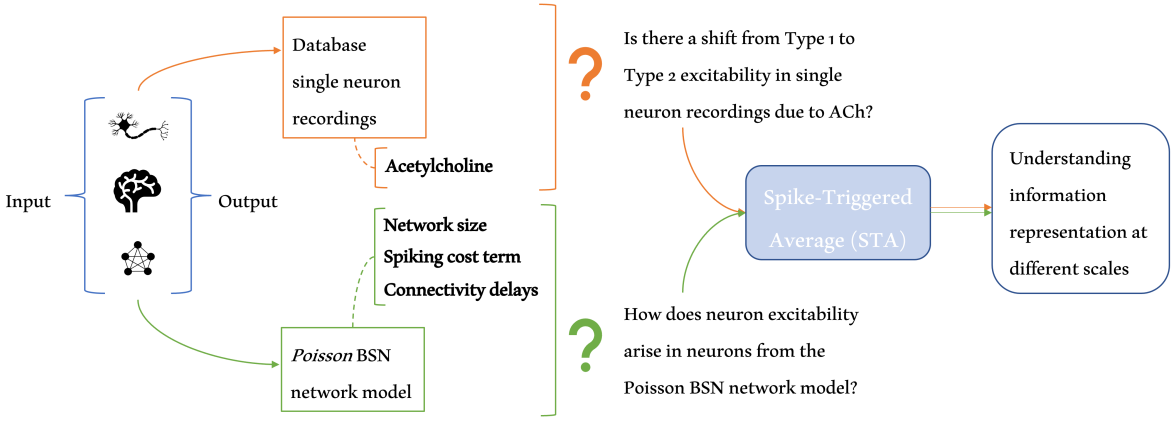

Figure 1: Overview of the project. Orange stream: Research question one and Chapter 1. Green stream: Research question 2 and Chapter 2. Blue boxes are shared core elements of the thesis.

Chapter 1

Cholinergic effect in the shift from type 1 to type 2 in a neuron

1.1 Background

How the brain represents information is still an open question. One approach to investigating this question consists of looking at the brain's coding properties. Such properties are centered in the stimulus-response relationship, defined by the relevant features of the input (stimulus) that can cause the output (response) [2]. Therefore, this stimulus-response dependency is based on just a few stimulus dimensions that relate to the response [16]. At a single neuron level, this dependency relates to the features of the input current (e.g., intensity or frequency) that make the neuron spike. The neuron's coding properties are thus centered on how neurons filter the input, and the closer the feature of the input is to that filter, the more likely the neuron will fire a spike. This idea of the filter has inspired neuron models such as the Linear-Nonlinear Poisson (LNP) models [17, 18], which will appear again in the second chapter.

There are two main neuron types that can be distinguished according to their coding properties and thus by how they represent information: type 1 and type 2 neurons [7]. These two types can be distinguished by what input features make the neuron spike. Type 1 neurons, also known as integrators, are sensitive to input intensity, so their firing rate (output) will increase if the input amplitude does too, and will decrease if the input amplitude also decreases. Contrary to that, type 2 neurons or resonators are sensitive to the input frequency. This means that their firing rate will be less variable compared to type 1 neurons and will be more specific for a given input frequency. This difference in selectivity comes mainly from the biophysical properties of the neurons' channels related to voltage potassium currents (I_M) [10, 11]. Nevertheless, these biophysical properties can be affected by neuromodulators by influencing channel gating related to these I_M currents [19]. But whether they can produce a shift from type 1 to type 2 in a single neuron remains largely unknown. ACh is at the center of attention concerning its potential role in causing a shift from type 1 to type 2 excitability in a single neuron [20] (for pioneer similar work, see also [19]). Namely, ACh is responsible for blocking the non-activating potassium channels, which decrease the flow of the I_M outward neuron's current [21]. In turn, it is hypothesised that this blockage is associated with the shift from type 2 to type 1 excitability, as it was first observed in the mouse visual cortex [19]. Therefore, the first research question of this thesis is: can ACh mediate a shift from type 1 to type 2 excitability in single neuron in vitro recordings from the mouse primary somatosensory motor cortex?

There are three well-established methods to assess whether a neuron is type 1 or type 2: Phase response curves (PRC), frequency-input (F-I) curves, and spike-triggered averages (STA) [9, 22, 23]. The nature of the injected input to the neuron will determine which of the three methods is optimal. PRC analysis is sensitive to a shift in phase of a constant spiking neuron given a perturbation, so the optimal input will be a constant current including a perturbation at a specific time. A type 1 PRC is monophasic (only positive values), and a type 2 PRC is biphasic (can take both positive and negative values). F-I curves, on the other hand, seek to relate input amplitude with firing rate. To do this, the injected input to the neurons has a step shape, the input's intensity increases in steps, and the correspondence with that intensity and firing rate is measured. Here, the F-I curve from type 1 neurons is continuous and shows an increased firing rate as the input amplitude increases, whereas for type 2 neurons, the firing rate of the neuron remains rather constant at a given input amplitude. Lastly, the STA method consists of averaging all the input traces of a small time length that come right before a spike. The resulting averaged input trace will be a representation of the stimulus feature that is responsible for triggering a spike. The optimal input of this method is the same or close to white noise. In the STA filters, type 1 neurons show a monophasic shape and type 2 biphasic. Regardless of the method, they can distinguish between type 1 and type 2 excitability. From a database of 43 neurons modulated with ACh and stimulated with a close to white noise input current [24], STA analyses will be applied to infer what input features made the neurons spike. From this analysis, we will assess whether there is a shift from type 1 to type 2 in each neuron and condition.

In the methods section of this chapter, the analyzed data will be shown (both input and recorded output), the STA method will be explained, and the data analysis procedure will be described. After, in the results section, the STA analysis of the experimental data will be shown. In the final section of this chapter, the results will be discussed.

1.2 Methods

To investigate whether single neural coding can change from type 1 to type 2 due to ACh, we have to study the input (current) and output (spike trains) relationship. In this section, we first describe the neural data used for the analysis. Then the STA analysis is explained. At the end of the section, there is a quick overview of the steps followed to obtain the results, relating the data with the STA analysis.

Experimental Data

The single neuron recordings used in this thesis come from a database of in vitro patch-clamp recordings of 232 cells from primary somatosensory and motor cortices (Layers 2/3) of adult mice (2-15 months old) of both sexes. During these recordings, the activation of serotonin, norepinephrine, acetylcholine, and dopamine receptors were regulated. Since the project aims to analyze the effect of ACh in single neuron excitability type, a total of 43 neurons modulated with ACh from this database were selected for this study (23 excitatory and 20 inhibitory). For more information regarding the experiments and database, see [24].

The input injected to stimulate the neurons was generated with the information transfer protocol [25]. This protocol generates an input with a wide range of current amplitudes (similar to white noise) for a short period of time, which avoids having incomplete recordings due to cell death. The input is based on a binary signal that alternates between 1 (on state) and 0 (off state) based on a Markov process, called the hidden state (Figure 1.1 **a**). Then the hidden state is transformed into a more biologically plausible input employing an artificial network. The resulting output of that artificial

network was the input injected to the soma of the neurons (Figure 1.1 **b**). This input is very close to white noise, but still preserves information about the hidden state, showing two types of correlations: the excitatory postsynaptic potential (EPSP) shape (of $5ms$) and the hidden state autocorrelation. The same input was injected into all neurons. The recorded output of each stimulation was the membrane potential of the axon (Figure 1.1 **c**).

Each neuron was stimulated 2 to 4 times, and although in each stimulation the input was the same, the conditions varied. From all recordings and neurons, four main conditions could be found. First, in the control condition, the medium where the neuron was stimulated was filled with artificial cerebrospinal fluid (ACSF), then the input current was injected and membrane potential recorded. This was the only condition present in all neurons. In the remaining three conditions, the ACh M1 receptor was regulated by adding to the ACSF medium an agonist (McA-N-343 (20 μM), second condition), antagonist (Pirenzepine dihydrochloride (5 μM), third condition), or both agonist and antagonist (fourth condition) (Figure 1.1 **d**). These conditions are key to distinguishing how ACh can influence what input features cause a neuron to spike and whether these correspond to type 1 or type 2 excitability.

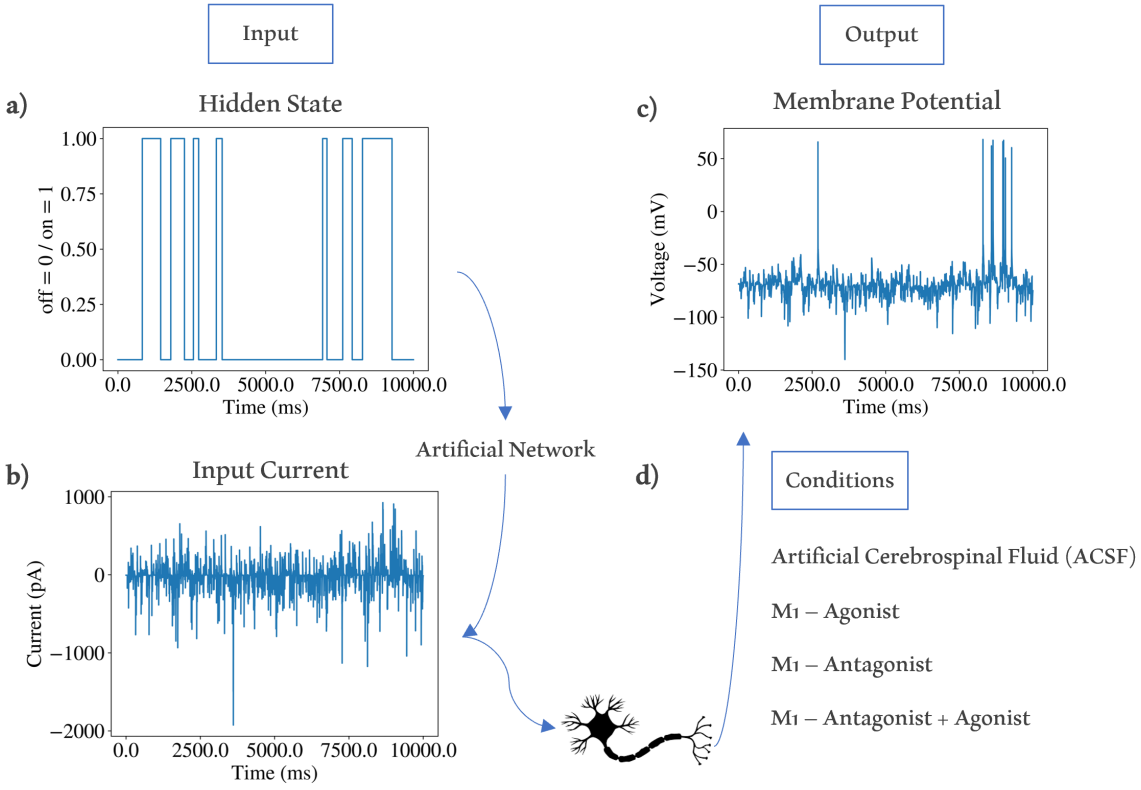

Figure 1.1: Example for one neuron of the steps followed to obtain input and output recordings. **a)** The hidden state: a generated signal of alternating 0 (off states) and 1 (on states). **b)** An artificial network converts the hidden state to a noisier signal which will be injected into the soma of the neuron. **c)** Recordings of the axon's membrane potential while the neuron is being stimulated. **d)** The input is the same for all recordings, only the stimulation conditions related to ACh change. All plots of this figure only contain data from the first 10 seconds of stimulation and recording from one neuron.

Spike-Triggered Averages

STA is a reverse correlation method that aims at selecting the relevant features of the time-varying input that caused the neuron to spike, establishing thus the stimulus-response relationship of the neuron. In other words, it is a dimensionality reduction from all possible input features to the key feature that triggered the spikes (for a thorough explanation of the STA, see [26, 27]). The underlying idea of this method is that if we randomly pick traces of the input of a short time length τ and average them, the resulting mean would be 0 and have a normal probability distribution $P(input)$ (this considering that the input has a zero mean). Contrary to that, if we select the input traces of time length τ that come right before the neuron spikes, we will have a non-zero mean vector with a probability distribution $P(input(t_i - \tau)|spike(t_i))$ centered on that mean. Here t_i are time indices of the spikes. These input trace vectors are the ones that supposedly trigger each spike, and the average of these can be seen as the feature of the input that makes a neuron spike. Hence, we can formalize the STA as:

$$STA = \frac{1}{N}\sum_{i=1}^{N} \vec{input}(t_i - \tau), \tag{1.1}$$

where N is the total number of spikes, t_i is the time of the i^{th} spike, and τ is the selected time window previous to the spike. The result of the STA is an averaged vector of length τ that corresponds to the first order (dimension) feature of the input that triggers the neuron to spike. This method is sensitive to distinguishing type 1 and type 2 excitability: in type 1 neurons, the resulting STA is a positive and monophasic filter, and in the STA from type 2 neurons the resulting filter is biphasic and can take positive and negative values [28, 29, 30].

Analysis Procedure

Each neuron from the database of [24] is recorded under one or more conditions (see *Experimental Data* section). STA analysis was done for each recording, so from each neuron, we could get as many STAs as recorded conditions in that neuron. Spike indices at times t_i from the membrane potential of each recording were matched with the input current at these same times t_i. Then τ ms length of the input trace previous to each spike was selected and normalised to have unit vector length. After, all the selected input traces were averaged. For this analysis, the time window τ was $10ms$. For a visualization of the STA analysis in one recording, see figure 1.2. The obtained (STA) filters are compared between the different conditions to investigate whether there is a shift from type 1 to type 2 excitability due to ACh effect.

The code to produce the analyses and figures in this Chapter can be found in: Data analysis ACh and plots.

1.3 Results

To study a shift from type 1 to type 2 excitability due to a cholinergic effect, a total of 43 neurons (23 excitatory and 20 inhibitory) were analysed. These neurons are from the primary somatosensory motor cortex (Layers 2/3) of adult mice (2-15 months old). Each neuron received first a control stimulation (ACSF), and after, one stimulation with at least one of the three other conditions (M1-Agonist, M1-Antagonist, or M1-Agonist + M1-Antagonist). In each recording, the membrane potential of the neurons was recorded, which was later used to determine the spike indices of the neuron in that condition. For all neurons and conditions, the recording time is 360 seconds. There are a total of

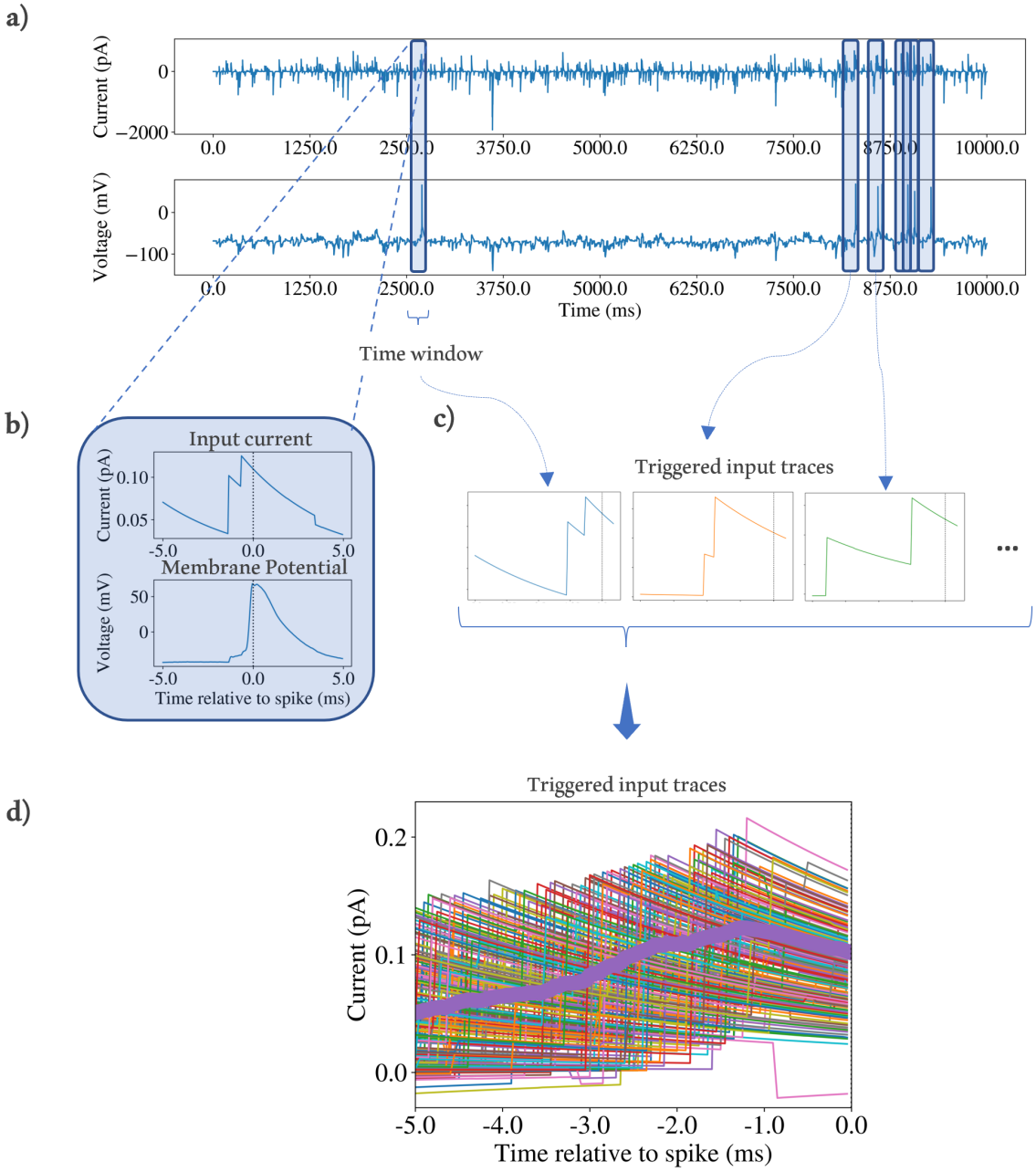

Figure 1.2: STA method applied in one recording. **a)** First, the input current and membrane potentials are aligned. All spikes from the membrane potential are found and indexed (blue boxes). **b)** A close-up of the correspondence between the input current (top panel) and the first spike (bottom panel). **c)** All input traces that triggered a spike are stored and normalized to unit vector length. **d)** The result of all the stored input traces that triggered a spike, and the thicker purple line is the average of all of them (the STA). This example comes from a recording of a type 1 excitatory neuron in the ACSF (control) condition.

123 recordings: 43 are stimulations with the ACSF condition (one per neuron), 42 stimulations with the M1-Agonist, 24 with the M1-Antagonist, and 14 with a combination of the M1-Agonist and M1-Antagonist. Neurons with unrealistic recordings of membrane potential values were eliminated from the analysis.

In figure 1.3 the STAs for all neurons and conditions are shown. In the ACSF condition, all recorded

neurons are type 1, which can be seen by the monophasic shape of the STA in panels **a** and **e**, for both excitatory and inhibitory neurons. In the M1-Agonist condition (**b** and **f** panels), the STA is also monophasic and only takes positive values, showing type 1 excitability for all neurons in that condition. This STA shape is also present in the rest of the conditions as well (**c**, **d**, **g**, and **h**), thereby showing no shift in excitability type due to ACh neuromodulation.

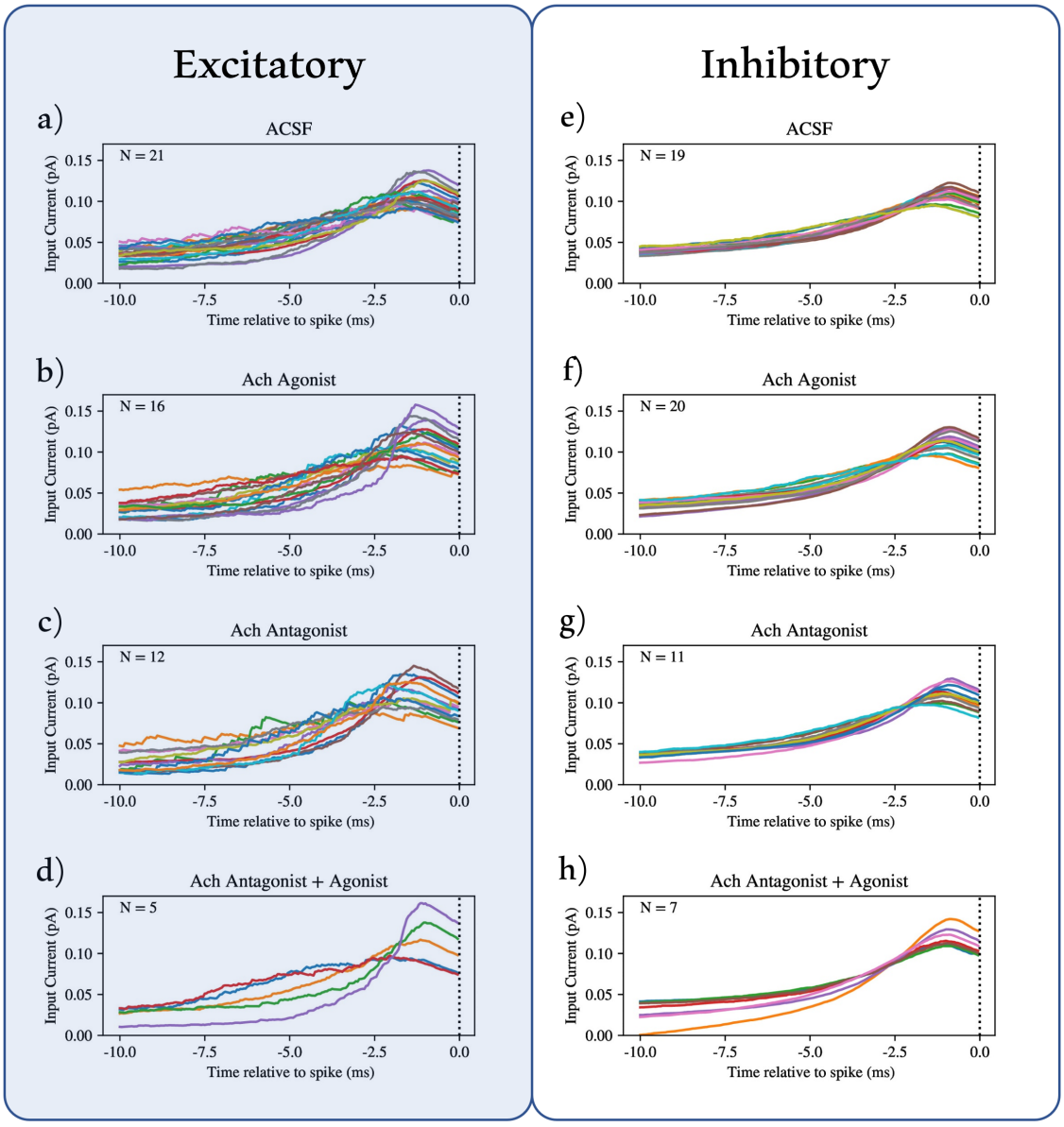

Figure 1.3: STAs of all recordings and conditions. The left column shows the results from excitatory neurons and the right column from inhibitory neurons. Each subplot of each column represents one condition (so two times four conditions, **a** to **d** and **e** to **h**). Every subplot has N number of STAs, which correspond to the number of neurons that were stimulated under a specific condition. The STAs of the ACSF control condition for excitatory and inhibitory neurons are shown in panels **a** and **e**. The STAs from the M1-Agonist condition in panels **b** and **f**. The STAs from the M1-Antagonist condition in panels **c** and **g**. Finally, all STAs from the condition M1-Agonist+Antagonist are shown in panels **d** and **h**.

1.4 Discussion

In this chapter, we have investigated whether ACh modulates a shift from type 1 to type 2 neural excitability from in vitro single neuron recordings from the primary somatosensory motor cortex of adult mice. This shift supposes a change of the coding properties of the neuron, from being sensitive to any input intensity (type 1 or integrator) to being only sensitive to a specific frequency (type 2 or resonator) [8]. Previous work hypothesized that ACh is responsible to modulate network activity based on the shift from type 1 to type 2 in single neurons [20]. This idea is based on a previous finding of a shift from the Phase Response Curves (PRC) of type 2 neurons to type 1 due to an ACh effect [19]. In this research, to measure the shift of excitability type due to ACh, STAs were computed for all neural recordings and conditions. We found that all 43 analyzed neurons (23 excitatory and 20 inhibitory) were type 1 before introducing ACh Agonists or Antagonists. Moreover, we did not find any shift in the STA in any neuron due to ACh effect. These results suggest that neither an ACh agonist nor antagonist can cause a shift from type 1 to type 2 excitability in type 1 neurons from the primary somatosensory mice cortex. In this study, we also did Principal Component Analysis (PCA) in our data, but we did not see any change in our results. That is why we did not include the PCA analysis in the thesis, and the main focus was put on the STA analysis.

The STA results obtained from the ACh-Agonist condition are in line with previous research from [19]. In their research, they found a shift from type 2 to type 1 excitability due to ACh agonist effects. The effect of an ACh agonist is to block potassium channels related to the I_M currents, and low I_M currents are related to type 1 neurons [19, 31]. Therefore, as the current research supports, type 1 neurons in the control (baseline) condition cannot shift to type 2 due to ACh agonist effects since the I_M currents are already low.

Interestingly, in in vitro extracellular recordings from the mouse somatosensory cortex, it was found that an ACh antagonist could potentiate neuron synchrony [32]. This indicates that no presence of ACh enhances type 2 excitability. Therefore, the ACh antagonist potentiates such synchrony since type 2 neurons are sensitive to specific frequencies from the input. This effect was also observed in modeled cortical type 2 neurons, where given low levels of ACh, the type 2-like PRC of such neurons got more accentuated [31]. However, in that research, the excitability baseline of such cortical models was type 2, so the antagonist ACh condition was not tested in type 1 cortical neurons. In our results, we saw that the presence of an ACh antagonist did not cause a shift from type 1 to type 2 neurons. Therefore, it seems that the ACh-antagonist effect is only present in neurons that are type 2 in baseline conditions.

The fact that no shift was found in the analysed data in this research, contrasts with the idea that type 1 cortical neurons can shift to type 2 excitability in neuron models [33] and in vitro experiments [11]. Namely, in such in vitro experiments, the slow voltage potassium current (I_M) was responsible for this shift. Yet, these changes were induced by current injected into the neuron, with no influence of ACh neuromodulation. ACh has a blocking effect in voltage potassium current, causing a shift from type 2 to type 1 excitability [19]. Although an increase of the I_M current could cause a shift from type 1 to type 2, given the obtained results in this research, it seems that the used ACh antagonists are not directly involved in such a shift. Hence, what causes an increase of the I_M current, which is responsible for the shift from type 1 to type 2 excitability, remains an open question.

Besides the actual neuron filter the STA represents, these filters are also shaped by the input correlation and the neuron's spiking history dependence. Concerning input correlation, STA filters will be unbiased if the input is uncorrelated, meaning that any input feature has the same probability of occurrence [17]. Nevertheless, STA can be biased if the input is correlated and therefore not be

an accurate representation of the STA filter [34]. To fix this, one can whiten the STA to correct such biases (for a detailed explanation see [34]). Whitening can amplify noise across all stimulus dimensions of the stimulus, and that is why the whitening correction should be further regularized. In this project, neither of these corrections were applied, so whitening and regularizing the STAs would be an improvement of the reliability of the analysis and results of this research. Regarding history dependence, the standard STA analysis assumes that the spike only depends on the stimulus and not on the history of previous spikes. This assumption might lead to inaccurate STAs since this analysis does not account for the neuron's refractory period or bursts [27]. To overcome the neuron's history dependence issue, one can only take the spikes that are preceded by a long interval of "silence" ($\sim$75ms) where no spike has occurred [26]. Hence, the analysis becomes an event-triggered average which avoids the spike history bias. This correction could also be an improvement of the analysis of the current investigation.

Despite the need for analytical improvements, STA without corrections can still describe the neural response although being sub-optimal [18, 26, 27]. Therefore, regardless of the highlighted improvements, the obtained results can still be considered valid. And with the subsequent corrections, the found results can shed light on a better understanding of the shift between type 1 and type 2 excitability in a single neuron due to ACh.

Chapter 2

Neural coding in Poisson Balanced Neural Networks

2.1 Background

The unknowns about information representation go beyond a single scale in the brain. A big challenge in neuroscience is to unify the observed dynamics and spike mechanisms from single neurons to explain information representation in neural networks. Indeed, the available techniques to record at a network scale are poor at explaining the relationship between network computations and single neuron dynamics [13]. In the previous chapter, we focused on information representation in single neurons. Here, the focus will be put on neural networks.

Theoretically derived network models can be very useful in explaining how neurons interact and relate to larger and more complex structures and computations seen in the brain. These models are very convenient because they can be studied on small scales (hundreds of neurons), have biologically plausible properties, and are easy to simulate. The balanced spiking networks (BSN) framework is one of the first models with such features [14]. The BSN model consists of coupled leaky integrate and fire (LIF) neurons, which aim at implementing a dynamical system or an autoencoder. This implementation is given by a tight balance between neurons that positively or negatively contribute to the output through their decoding weights. The network's output is therefore the filtered spike trains of all neurons with their respective decoding weights. The BSN framework has at least two main inconveniences regarding its biological plausibility: deterministic thresholds and almost instantaneous connectivity between neurons. The *Poisson* BSN model [15] is based on the BSN model but accounts for non-deterministic thresholds and includes delays to neural connectivity.

In the *Poisson* BSN model (as well as for the BSN), neural spiking is irregular, the network is robust to perturbations and neural loss, and each spike contributes very efficiently to the output of the network [14, 15, 35]. Moreover, the LIF neurons used to implement these network models fall into the Linear-Nonlinear Poisson (LNP) neuron model [17]. This model has three components: one is a linear filter, which selects the relevant features from the input that the neurons receive. The second component is the non-linearity used to transform how well the filter matches the input to a firing rate. And lastly, spikes are randomly generated using a Poisson process on the firing rate. Therefore, the coding properties of the LIF neurons that form the *Poisson* BSN network depend on their filters. This means that their probability of spiking will be related to a selective feature of the input. On top of that, the *Poisson* BSN model has three parameters (network size, spiking cost term, and time delays)

that can influence the spiking pattern of the neurons, and thereby the input-output relationship. First, network size will determine the number of spikes of a single neuron, making it more or less efficient. Second, the spiking cost term (or refractory period) punishes repetitive firing from a firing neuron, and third, the connectivity delays affect the neurons' voltage updates conditioning their firing timings. However, despite the number of elements that might influence the coding properties of the neurons in the *Poisson* BSN model, how such properties arise and interact at a network level remain unknown. Therefore, the second research question of this thesis is: how does the stimulus-response relationship of the *Poisson* BSN neurons arise and interact due to tuning the network size, spiking cost term, and connectivity delays?

In this chapter, the *Poisson* BSN model will be implemented, and the same STA analysis from Chapter 1 will be applied to the neurons of this model. The STA allows us to study how input and output relate to each neuron in the context of a neural network. By changing the parameters of the network, we will see how the coding properties arise in these networks and how they can change and interact when modifying the network properties.

In the methods section of this chapter, the *Poisson* BSN model will be explained. The network parameters and input will be shown, along with the applied analyses to assess network performance (STA, network error, and network efficiency). The outcome of the analyses from all simulations will be shown in the results section. The chapter ends with a discussion of the obtained results.

2.2 Methods

We used the *Poisson* BSN model ([15]) to investigate how the coding properties of the neurons might arise and be affected by the interaction with other neurons at a network model level. In this section, we go over the main points of the derivation of the network, we show the simulation parameters, and finally, explain the outcome of each simulation.

Poisson Balanced Spiking Network model

The *Poisson* BSN model [15] is based on the BSN model [14]. We will first explain the BSN network, and continue with the updates that the *Poisson* BSN model suggests. For an illustrated overview of the *Poisson* BSN model see figure 2.1.

The goal of both BSN and *Poisson* BSN models is to implement a spiking neural network that can autoencode a given input $\mathbf{i}(t) \in \mathbb{R}^J$. For such an autoencoder implementation, the network aims to reproduce as accurately as possible the input. Therefore, the target output of the network $\mathbf{x}(t)$ is equal to $\mathbf{i}(t)$. The first assumption of this model is that the spike trains can be linearly decoded. Therefore the network's approximation $\hat{\mathbf{x}}(t)$ of the target output $\mathbf{x}(t)$ is the weighted combination of the filtered spike trains of all neurons:

$$\hat{\mathbf{x}}(t) = \mathbf{D}\mathbf{r}(t), \tag{2.1}$$

where $\hat{\mathbf{x}}(t) \in \mathbb{R}^J$ is the estimate or approximation of the network, $\mathbf{D} \in \mathbb{R}^{J \times N}$ are the decoding weights, which are equally divided into positive and negative ones, and $\mathbf{r}(t) \in \mathbb{R}^N$ are the filtered spike trains. J stands for stimulus dimensions and N for the number of neurons in the network. The filtered spike train $\mathbf{r}(t)$, is a convolution of each neuron's spike train $s_i(t)$ and an exponential decaying function $g(t)$. So for the i^{th} neuron of the network we define the filtered spike trains as:

$$r_i(t) = s_i(t) * g(t) \tag{2.2}$$

Now, the network aims to closely reproduce the target output $\mathbf{x}$. To do so, the network keeps track of the squared error between its estimate $\hat{\mathbf{x}}(t)$ and the target output $\mathbf{x}$. With this, the network tries to minimize the error function:

$$E(t) = ||\mathbf{x}(t) - \hat{\mathbf{x}}(t)||_2^2 \tag{2.3}$$

This error will determine whether the neurons of the network should spike or not, which brings us to the second assumption of this model: neurons are only allowed to spike if they reduce the error function. Since the estimate $\hat{\mathbf{x}}(t)$ is given by a linear decoding of the filtered spike train ($\hat{\mathbf{x}}(t) = \mathbf{D}\mathbf{r}(t)$, first assumption), the spikes can directly change the estimate and therefore allow for a better approximation of the target $\mathbf{x}(t)$. With this, we find that a neuron will spike if

$$||\mathbf{x_t} - \hat{\mathbf{x_t}}||_2^2 < ||\mathbf{x_t} - (\hat{\mathbf{x}}_\mathbf{t} + \mathbf{D_i})||_2^2 \tag{2.4}$$

We see that the weight $\mathbf{D_i}$ of the i^{th} neuron is the contribution it has to the readout $\hat{\mathbf{x}}(\mathbf{t})$ at time t. Expanding this inequality we reach

$$\frac{\mathbf{D_i}}{2} < \mathbf{D_i^T}(\mathbf{x_t} - \hat{\mathbf{x_t}}) \tag{2.5}$$

The term on the left hand side is defined as the threshold error, so the i^{th} neuron will spike when a maximum error is reached, given by that threshold. The term on the right hand side is defined as the voltage of the neuron, which increases or decreases based on how different the target output $\mathbf{x}(t)$ and the network readout $\hat{\mathbf{x}}(\mathbf{t})$ are throughout time. So for clarity, we can rewrite the threshold T_i and voltage v_i equations:

$$T_i = \frac{\mathbf{D_i}}{2}, \text{ and } v_i = \mathbf{D_i^T}(\mathbf{x_t} - \hat{\mathbf{x}}_\mathbf{t}) \tag{2.6}$$

So far, we have described the general derivation of the BSN model [14]. One of the key features of the *Poisson* BSN model [15] is that the spiking is not restricted to a deterministic threshold as we are describing here with T_i. Therefore, the *Poisson* BSN model includes a probabilistic term called conditional intensity. This term determines the probability of a neuron to spike λ_t at any small time window Δ. The conditional intensity of every neuron λ_t is a function of the neuron's voltage. Hence, at every small time window, the neuron has an independent probability of spiking that changes over time. This λ_t function has a sigmoid shape, and the lower and higher bounds are determined by the maximum and minimum firing rates of the neuron

$$\lambda_t = f(v_t) = \frac{F_{max} - F_{min}}{1 + e^{-\alpha(v_t - T_i)}} + F_{min} \tag{2.7}$$

Here v_t stands for the membrane potential of one neuron at time t (right hand side of equation 2.6), F_{max} and F_{min} are the maximum and minimum firing rates, and T_i is the threshold of the i^{th} neuron (left hand side of equation 2.6). Parameter α determines the slope of the sigmoid. In other words, higher α values will make the threshold more deterministic (steeper slope), and alternatively, lower α will make the spiking more probabilistic (flatter slope). By letting α and F_{max} tend to ∞, we would get a totally deterministic threshold (figure 2.1 **b**).

The conditional intensity λ_t is bounded by the maximum and minimum firing rates. Therefore, to convert these to a spiking probability, the following spike rule is applied:

$$P(s_t = 1|\lambda_t) = 1 - exp(-\Delta\lambda_t) \tag{2.8}$$

where s_t stands for a spike at time t, and the term $exp(-\Delta\lambda_t)$ is the probability of not seeing a spike. This probability of spike per neuron allows more than one neuron to spike at the same time bin Δ.

Another element that is still missing is the delays of neural connectivity. In the *Poisson* BSN network model, all neurons connect to each other. So if a neuron spikes, the overall error $E(t)$ of the network is reduced, and in turn, this causes two effects on all neurons. One, the neuron that spiked self-resets its voltage. Second, via the decoding weights $\mathbf{D}$, the voltages of all neurons are updated through the recurrent connectivities. In the *Poisson* BSN model, a delay can be included in these recurrent connectivities (see figure 2.1 **a**).

Moreover, a quadratic cost term μ is added to the error function to penalize high spiking activity and enhance spike sparseness throughout the network. Therefore, the error function from equation 2.3 becomes:

$$E(t) = ||\mathbf{x} - \hat{\mathbf{x}}||_2^2 + \mu||\mathbf{r}(t)||_2^2 \tag{2.9}$$

and if we expand the error function, now the threshold and voltages from equation 2.6 are as follows:

$$T_i = \frac{\frac{\mu}{\tau^2} + ||\mathbf{D_i}||^2}{2}, \text{ and } \mathbf{v_t} = \mathbf{D^T}(\mathbf{x_t} - \hat{\mathbf{x_t}}) - \frac{\mu}{\tau}\mathbf{r_t} \tag{2.10}$$

The term τ represents the voltage leak and is added for biological plausibility as done in [14]. If we derive the voltage equation from 2.10, the spiking cost term shows up as a spiking reset cost (see equation 2.12), which penalizes repetitive firing and is relatable to the refractory period of the neuron.

With this, at each time step, the network model updates the voltages of all neurons according to the error tracking between $\hat{\mathbf{x}}$ and $\mathbf{x}$. It also computes the vector of the conditional probabilities λ_t for all neurons, which is converted to a probability of spiking for each neuron. If a neuron spikes, the self-reset occurs at the following time bin, and the updates from the recurrent connectivities will occur after the determined delay d.

For each time step Δ of the simulation of the *Poisson* BSN network, we end up with five computations in each neuron. We use forward Euler to simulate the following differential equations derived from the equations above. The first computation is an update of the filtered spike trains $r(t)$:

$$\dot{r}_i = -\frac{1}{\tau}r_i(t) + s_i(t) \tag{2.11}$$

here τ is the rate of the exponential voltage decay, $\dot{r}_i$ is the rate of change in the filtered spike train at time t. The second computation is an update of the voltage $v(t)$ of all neurons such that:

$$\dot{v}_i = -\frac{1}{\tau}v_i(t) + \mathbf{D^T}(\frac{1}{\tau}\mathbf{x}(t) + \dot{\mathbf{x}}(t)) - (\mathbf{\Omega}_{self}(t) + \mathbf{\Omega}_{rec}(t-d)) \tag{2.12}$$

where $Dotv_i$ is the voltage's rate of change, $\mathbf{D^T}$ is the transpose of the decoding weight matrix of all neurons, $\mathbf{x}(t)$ and $\dot{\mathbf{x}}(t)$ are the target output and its derivative respectively, and d stands for the delay in the recurrent connections $\mathbf{\Omega}_{rec}$. The term $\mathbf{\Omega}_{rec}$ is the matrix multiplication $\mathbf{D^T D}$ (excluding the diagonal) and represents the voltage update from one neuron to the others. Additionally, $\mathbf{\Omega}_{self}(t)$ is the self-update of the neuron and is defined by $\text{diag}(\mathbf{D^T D}) + (\mu/\tau^2)$. μ is the quadratic cost term, and τ is the exponential decay rate of the neuron. The third computation is the calculation of the conditional intensity seen in equation (2.7), and the probability of a spike (equation 2.8) is the fourth computation. Finally, a random process is realized to determine whether the neuron will spike or not.

For a thorough analysis and derivation of the equations of the BSN and *Poisson* BSN models see [14] and [15] respectively.

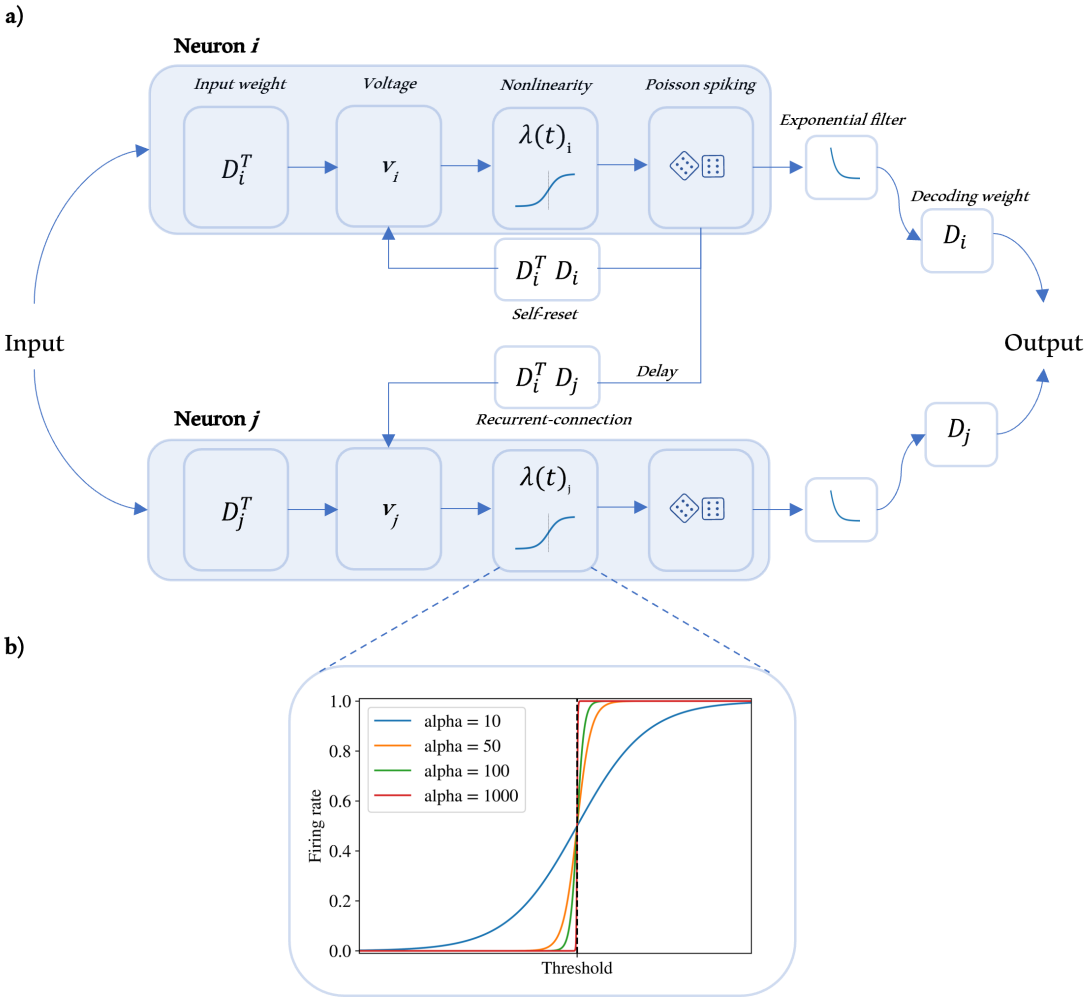

Figure 2.1: Scheme of the *Poisson* BSN model. a) A two-neuron simplification of the model. Each blue box represents one neuron, and the smaller boxes inside are the different components of the computations that form it. Through the input weights D_i^T, the input changes the membrane potential v_i of each neuron. The voltage goes through a non-linearity (see panel **b**) that determines the probability of spiking ($\lambda(t)_i$). Spikes are filtered by an exponential filter and decoded by D_i. The contribution of all neurons forms the output ($\hat{X}$). Only the self-reset for neuron i is shown, and only recurrent connection from neuron i to neuron j is shown.**b)** The slope of the conditional intensity $\lambda(t)_i$ is determined by the parameter alpha of equation 2.7. This figure is an adaptation of the scheme that can be found in [15]

Simulation Parameters

In Table 2.1 the simulation parameters used throughout all simulations are shown.

Input

The input given to the network is the same as the one used to stimulate the neurons during the patch-clamp experiments explained in the previous chapter (see figure 1.1 **b**). This input is a transformation from a signal of ones and zeros (hidden state) to an almost white noise signal. For every stimulation, the input is the same.

	Parameter (units)	Value	Range
$\|D\|$	absolute value decoding weight	1	
σ	SD noise weights	0	
$\frac{1}{\tau}$	decay time constant (ms)	20	
T	time simulation (s)	25	
Δ	time step (ms)	0.001	
F_{max}	maximum firing rate (mHz)	20	
F_{min}	minimum firing rate (mHz)	0	
α	conditional intensity slope	10	
N	network size	20	[2, 400]
μ	spiking cost term (ms)	0	[0, 5]
d	time delay (ms)	0	[0, 1]

Table 2.1: Simulation parameters.

Network Performance Analysis and Statistics

Every simulation involved a change of parameters: network size, spiking cost terms, and time delays. The network size parameter is central in this investigation since we are interested in seeing how neurons interact and affect their coding properties in a network context. The other two parameters, spiking cost term and time delay, either affect the self-connections or the recurrent connections respectively. Therefore, they are relevant to neural interaction and have the potential to modify the coding properties of neurons at a network level. To study how the coding properties of neurons arose and interacted at a network level, the STA analysis was used exactly as described in the methods section of the previous chapter (see equation 1.1). On top of that, three further analyses in every simulation were made so that they could be related to the parameter tuning and the coding properties of the neurons. First of all, the Mean-Squared Error (MSE) has been computed in every simulation, comparing the readout $\hat{X}$ with the target output X at every time step of the simulation:

$$MSE = \frac{1}{T}\sum_{t=0}^{T}(\hat{X}_t - X_t)^2 \tag{2.13}$$

Where T is the total time steps of the simulation. Moreover, network efficiency in terms of the mean number of spikes and standard deviation of all neurons (N) in each simulation was measured:

$$Mean = \frac{\sum spikes}{N} \tag{2.14}$$

$$SD = \sqrt{\frac{\sum |spikes - Mean|^2}{N}} \tag{2.15}$$

A software implementation of the *Poisson* BSN model and code to produce the figures in this Chapter can be found in : Network modelling analysis and plots

2.3 Results

To investigate how coding properties arise and interact at a network level, the *Poisson* BSN model was implemented and three main parameters were independently varied: network size, spiking cost terms, and time delays. Moreover, interactions between the parameters were also studied. In this section, we show how each parameter affected network performance and the coding properties of the neurons (namely, their STAs). In each section, we display the results of a few simulations in which we only

modify one parameter gradually while the rest remain constant. In the final section, the interaction between the parameters is shown.

Network size

By increasing the network size from 2 neurons up to 100, we could visually see a significant improvement in how the network can estimate the target output $\mathbf{x}(t)$ (figure 2.2 **a**). The raster plots (figure 2.2 **b**), show how the spiking of every neuron in the network ($N = 100$) becomes more efficient with less spikes per neuron. These differences were also seen when the network size in each simulation was modified ($N = 2$, 10, 25, 50, 100, 200, and 400) while spiking cost terms and time delays remained 0 throughout the simulations (see panel **c**). On the left plot of panel **c**, up until around 200 neurons, the more neurons in a network the more efficient each neuron becomes (fewer spikes). However, at a network level, the larger the size the more total number of spikes (less efficiency at a network level) (center plot of panel **c**). Increasing the number of neurons over 200, the number of spikes in single neurons became higher again. Larger networks showed less error in estimating the target output. The STAs also showed changes when modifying the network size parameter. For very small networks (e.g., $N = 2$ or $N = 10$), the filter of the neurons was monophasic. On the other hand, from around $N \approx 25$, the STAs increasingly showed a biphasic shape. Also, the larger the network size, the more differentiated the STAs of the positive and negative weights were. However, they still both had a biphasic shape.

Cost term

The spiking cost term is used to penalize a neuron if it has recently spiked. A very high spiking cost term (e.g., 5ms) drastically reduced the number of spikes of the network (see raster plot in figure 2.3 panel **b**), which also resulted in a poorer estimation of the input (figure 2.3 panel **a**). Up to a cost term of around 1ms, although there is a reduction of spikes, the MSE did not change, which meant that the network is more efficient (both for a single neuron and the total network spiking) without losing accuracy (figure 2.3 panel **c**). However, higher spiking costs penalized so much the spiking that the network started showing more errors in the tracking of the input (cost term $\approx$ 1ms). Changing the cost term values also affected the shape of the STAs (figure 2.3 **d**). Namely, for a very low spiking cost, the STAs in a network of 20 neurons showed a biphasic shape. Yet, as the cost term increased, the filters became more monophasic, being sensitive to only positive (or negative for the neurons with negative weights) parts of the input.

Time delays

The delays applied in these simulations referred to the recurrent connections (see figure 2.1), so after one neuron had spiked, the update to the rest of the network was delayed. In figure 2.4, from panel **a**, we could see that already a delay of 0.5 ms caused a worse estimation $\hat{\mathbf{x}}(t)$ of the target output $\mathbf{x}(t)$. The spike count also increased, and spiking became more synchronous between neurons (panel **b** from figure 2.4). By increasing the time delay the number of spikes also increased (in single neurons and at a network level), as well as the MSE of the network (panel **c**). These changes also came with differences in the STA filters depending on the delay (figure 2.4, panel **d**). From 0 to 0.5 ms the biphasic shape of the STA got accentuated. However, delays longer than 0.5 ms caused an increasing sinusoidal shape of the STA.

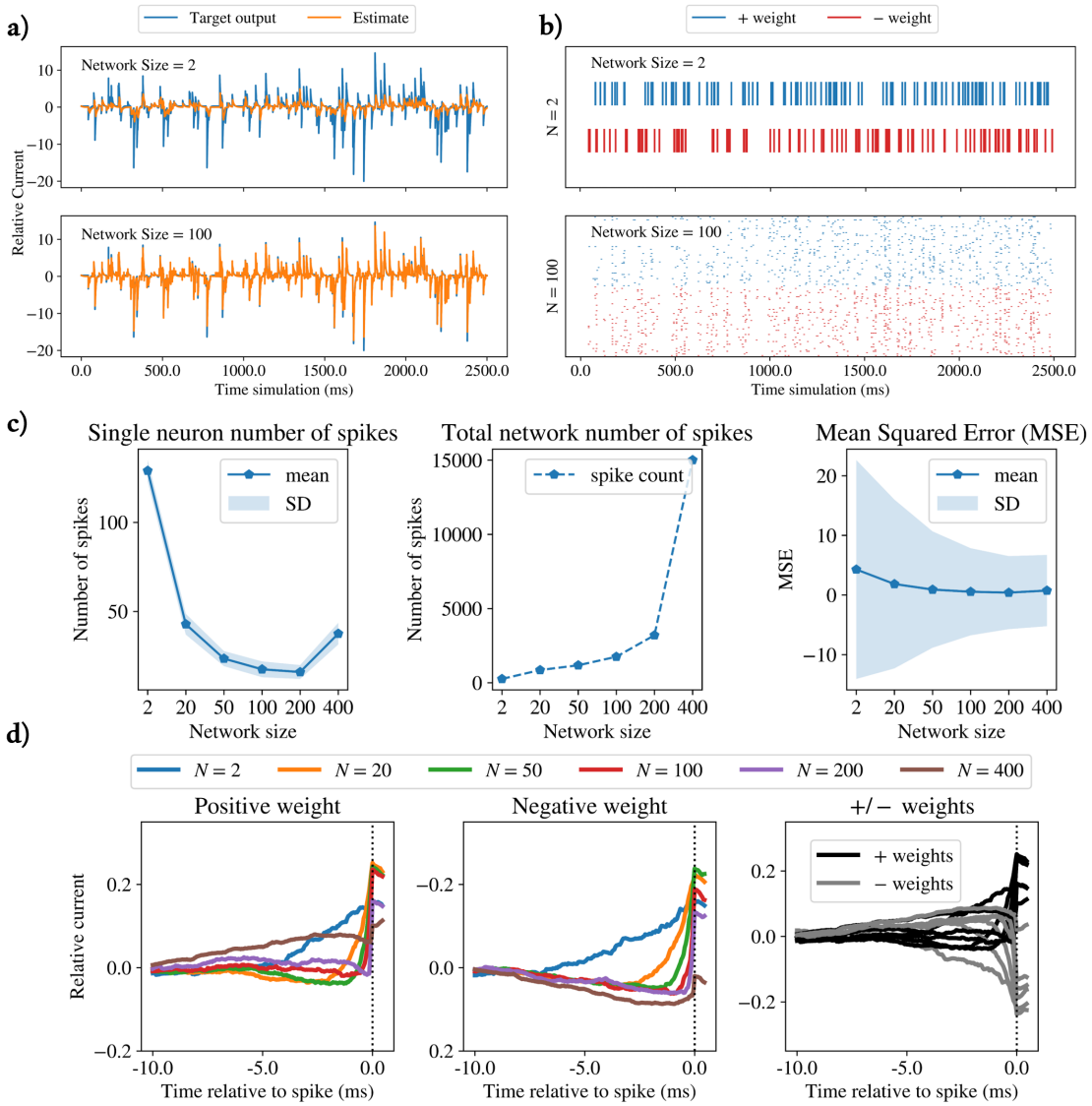

Figure 2.2: Effect of network size on the simulations of the *Poisson* BSN model and STA analysis. **a**) Representation of the target output $\mathbf{x}(t)$ (blue line) and estimate $\hat{\mathbf{x}}(t)$ (orange), the top plot show the result of network size $N = 2$, and bottom $N = 100$. **b**) Raster plots of the two network size conditions $N = 2$ (top) and $N = 100$ (bottom). The blue color represents the spikes of positively weighted neurons and the red of neurons with negative weight. **c**) Left: Average number of spikes per neuron depending on the network size, and the shaded areas are the standard deviations. Center: total number of spikes of the network depending on its size. Right: MSE of the difference between target output X and estimate $\hat{X}$ based on each network size value. **d**) STAs of the neurons depending on the network size. The left panel represents the average of the STA filters of the neurons with positive weight, and each color (simulation) represents a different network size. The center plot shows the average STAs of the negatively weighted neurons of the different conditions. Note that the vertical axis of this plot was shifted. The plot on the right shows all STAs of both positively and negatively weighted neurons.

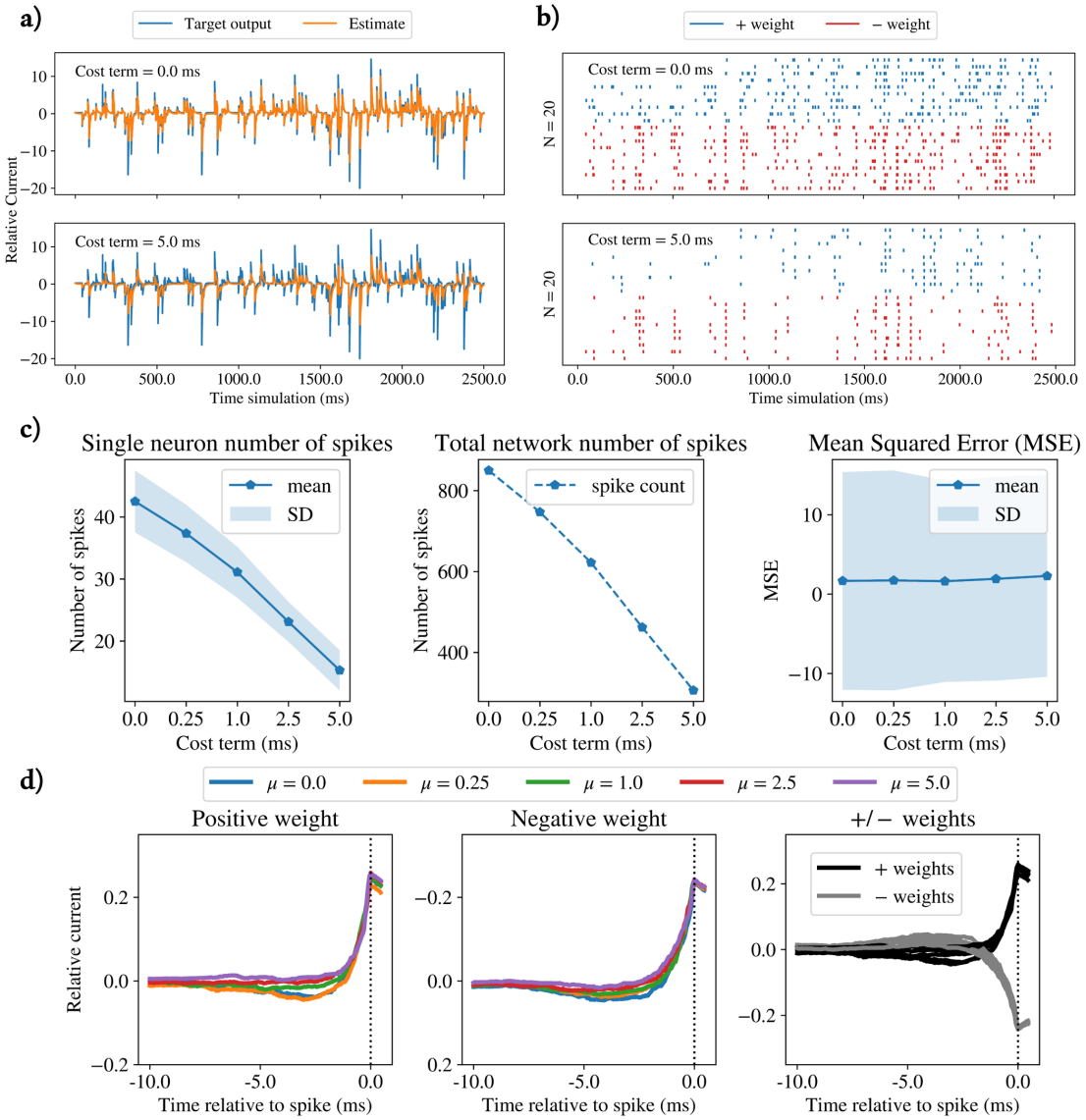

Figure 2.3: Effect of spiking cost term (μ) in *Poisson* BSN model and STA analysis. **a**) Representation of the target output $\mathbf{x}(t)$ (blue line) and estimate $\hat{\mathbf{x}}(t)$ (orange), the top plot shows the result of a cost term $\mu = 0$, and bottom $\mu = 100$. **b**) Raster plots of the two cost term conditions $\mu = 0ms$ (top) and $\mu = 5ms$ (bottom). The blue color represents the spikes of positively weighted neurons and the red of neurons with negative weight. **c**) Left: Average number of spikes per neuron depending on the cost term, and the shaded areas are the standard deviations. Center: total number of spikes of the network depending on the cost term. Right: MSE of the difference between target output $\mathbf{x}(t)$ and estimate $\hat{\mathbf{x}}(t)$ based on each cost term value. **d**) STAs of the neurons depending on the cost term. The left panel represents the average of the STA filters of the neurons with positive weight, and each color (simulation) represents a different cost term. The center plot shows the average STAs of the negatively weighted neurons of the different conditions. Note that the vertical axis of this plot was shifted. The plot on the right shows all STAs of both positively and negatively weighted neurons.

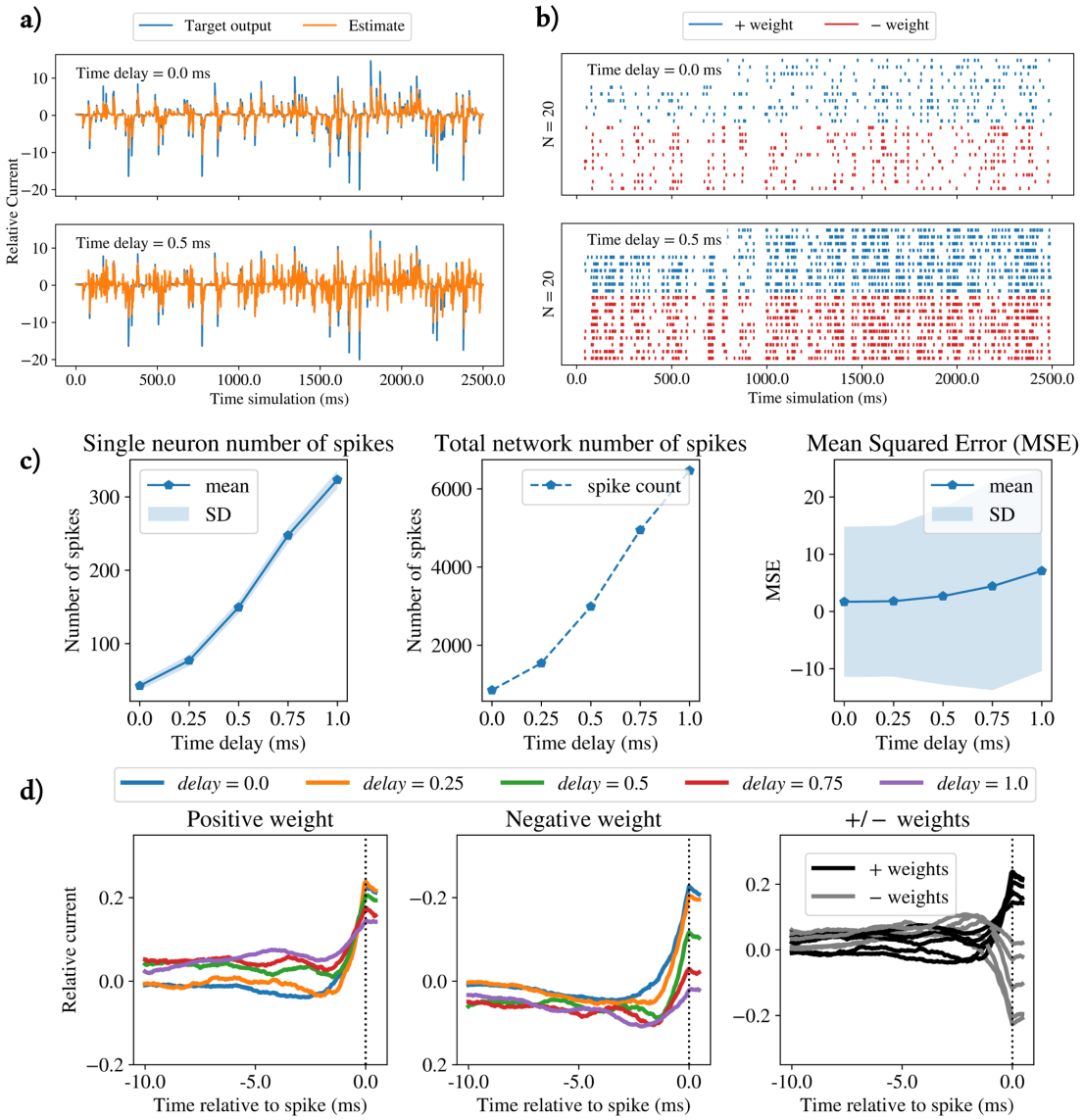

Figure 2.4: Effect of time delays in the recurrent connectivity on the simulations of the *Poisson* BSN model and STA analysis. **a**) Representation of the target output X (blue line) and estimate $\hat{X}$ (orange), the top plot shows the result of a $delay = 0ms$, and bottom $delay = 0.5ms$. **b**) Raster plots of the two delay conditions $delay = 0ms$ (top) and $delay = 0.5ms$ (bottom). The blue color represents the spikes of positively weighted neurons and the red of neurons with negative weight. **c**) Left: Average number of spikes per neuron depending on the delay, and the shaded areas are the standard deviations. Center: total number of spikes of the network depending on the delay. Right: MSE of the difference between target output $\mathbf{x}(t)$ and estimate $\hat{\mathbf{X}}(t)$ based on each delay value. **d**) STAs of the neurons depending on the delay. The left panel represents the average of the STA filters of the neurons with positive weight, and each color (simulation) represents a different delay. The center plot shows the average STAs of the negatively weighted neurons of the different conditions. Note that the vertical axis of this plot was shifted. The plot on the right shows all STAs of both positively and negatively weighted neurons.

Interaction between parameters

Three parameter interactions were studied: network size and cost term (figure 2.5), network size and time delay (figure 2.6), and cost term and time delay (figure 2.7).

To test the interaction between network size and spiking cost term, simulations of the *Poisson* BSN were made while changing these two parameters. Network size had a general effect of reducing the

spike count average and the MSE of the network (see figure 2.5). The spiking cost term drastically reduced the spike number average for small networks (> 20) but had less effect in larger networks. Yet, regardless of high or low spiking cost term values, the MSE of the network was only affected by its size (also see figure 2.5). As it is displayed in figure 2.2, the network size made the shape of the STA more biphasic for higher N up to around $N = 200$. This effect was countered by the cost term, where higher cost term values changed the STA shapes to monophasic also in large network sizes regardless of the valence of the neurons' weights (figure 2.5 **b**).

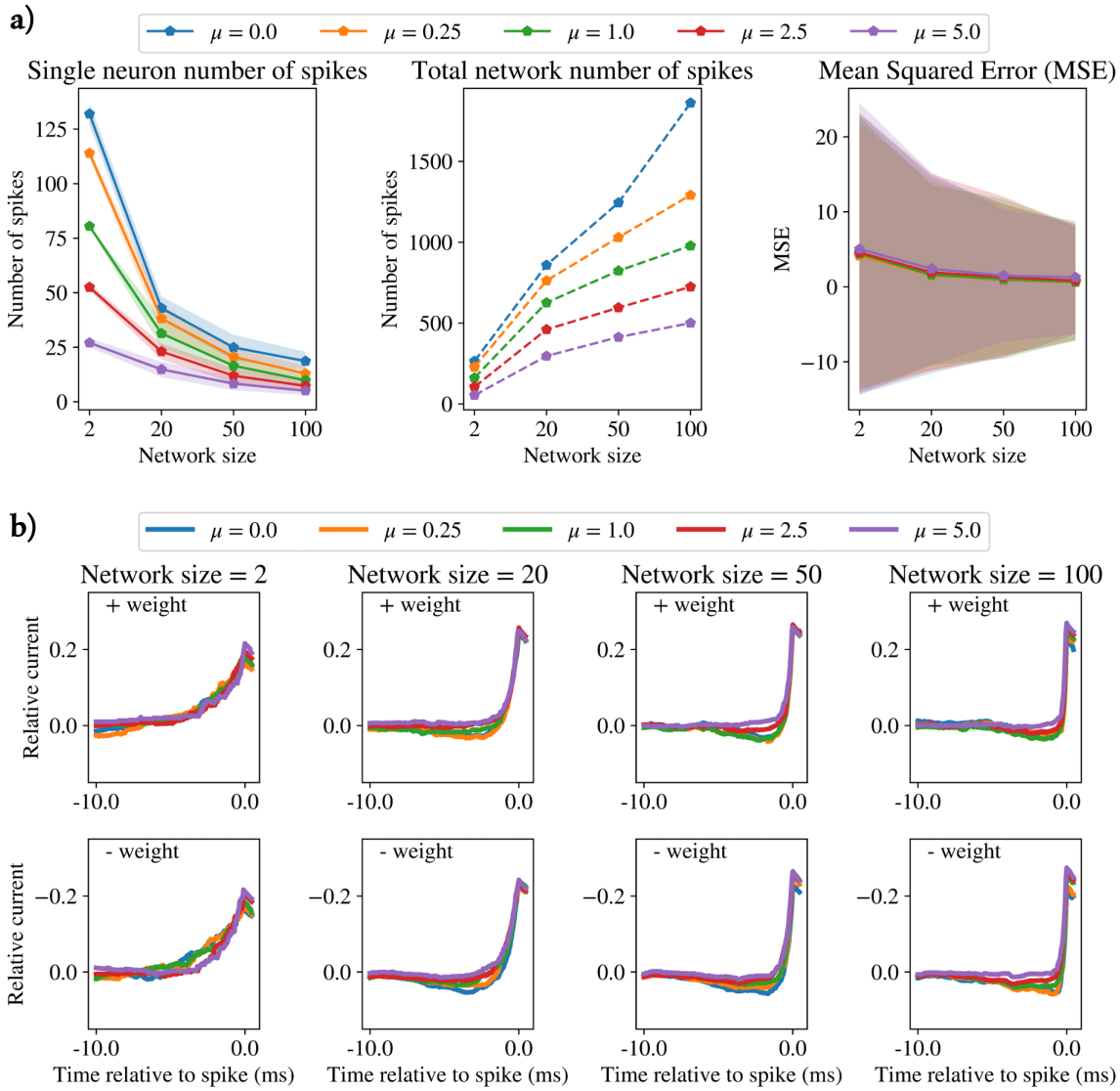

Figure 2.5: Interaction between network size and cost term (μ). **a**) Left: The average number of spikes for a single neuron, the shaded areas are the standard deviations. Center: Total number of spikes of the network. Right: MSE between the target output $\mathbf{x}(t)$ and estimate $\hat{\mathbf{x}}(t)$, the shaded areas are the standard deviations. Each color represents a different cost term value, which also applies for panel **b**. **b**) STA filters with different network sizes and cost term parameters. Every column represents a different network size. In the top row, the STAs of all positively weighted neurons are plotted, and the neurons with negative weight are plotted in the bottom row. In the bottom row, the y axis of each plot is shifted.

The interaction between a change in time delay and network size showed to change network performance and the input-output relationship of the network's neurons. For small network sizes ($N < 10$),

time delays did not affect the average number of spikes of the network nor its error. However, in larger network sizes ($N > 10$) both the average number of spikes and MSE were very affected by longer delays (figure 2.6 **a**). The STAs were also affected by the interaction between time delays and network size. Again, for small network sizes ($N < 10$), the STAs did not change regardless of the delay of the recurrent connections between neurons (up to 1 ms), so for a network of $N = 2$ the STA remained monophasic and for a network of $N = 10$ the STA remained biphasic. However, time delays did affect the shape of the STA filters for larger network sizes ($N > 10$), and the larger the network, the more affected by the delay (figure 2.6).

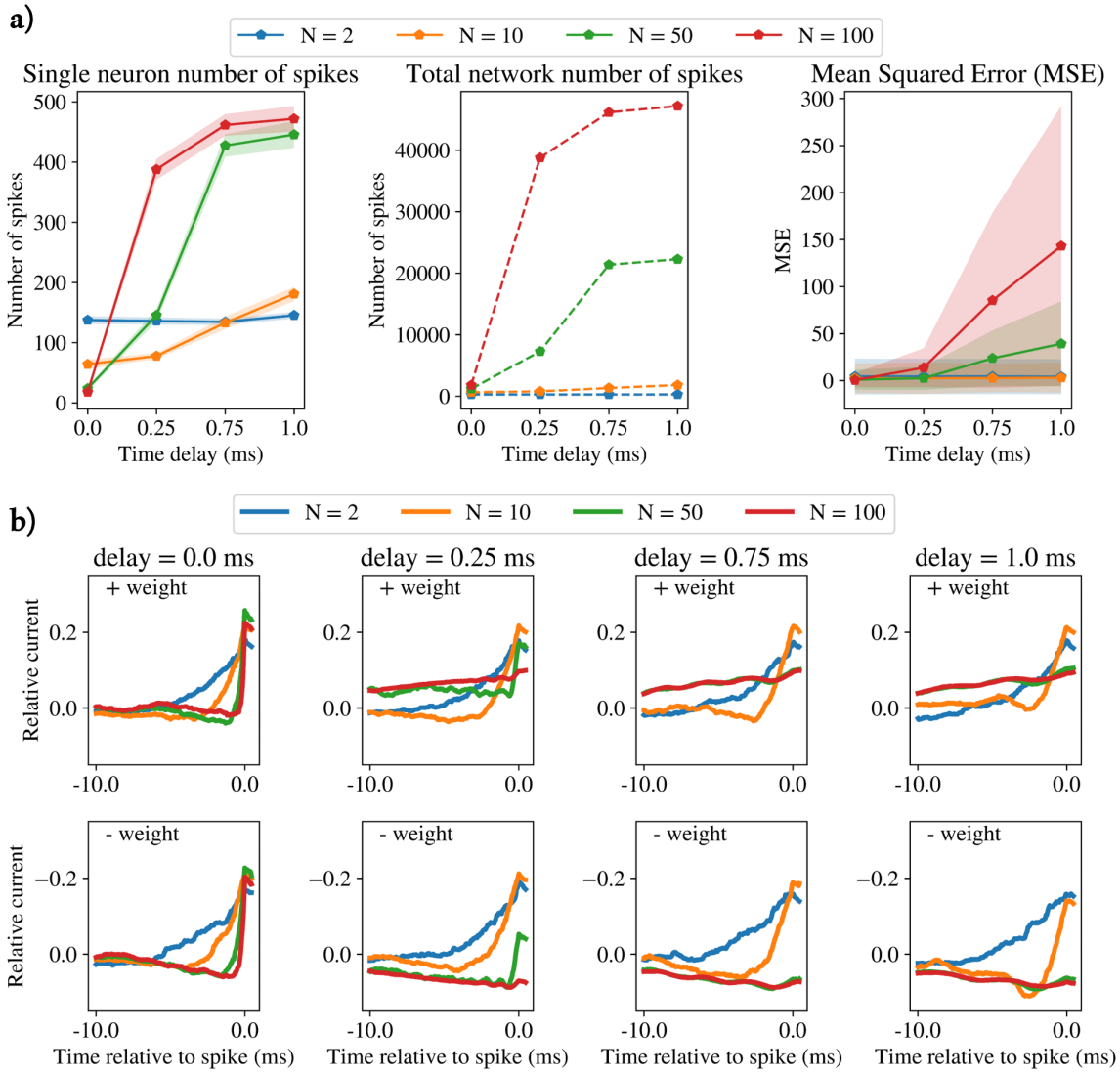

Figure 2.6: Interaction between time delays and network size. **a**) Left: The average number of spikes for a single neuron, the shaded areas are the standard deviations. Center: Total number of spikes of the network. Right: MSE between the target output $\mathbf{x}$ and estimate $\hat{\mathbf{x}}$, the shaded areas are the standard deviations. Each color represents a different network size value, which also applies for panel **b**. **b**) STA filters with different time delays and network size parameters. Every column represents a different time delay. In the top row, the STAs of all positively weighted neurons are plotted, and the neurons with negative weight are plotted in the bottom row. In the bottom row, the y axis of each plot is shifted.

Finally, in a network of 50 neurons, the combination of time delays and spiking cost also affected

network performance and coding properties of the neurons of that network. Although an increasing time delay ($delay \geq 0.25$) affected negatively in higher spike number averages and network error, by adding a cost term these effects were cancelled (figure 2.7 **a**). By increasing the time delays but having no cost term, the STAs of the neurons (both with positive and negative weights) become increasingly biphasic until a point where they got flattened with a small slope around 0 (reffig.10 **b**). Nevertheless, by increasing the cost term while increasing time delay (up until $delay = 0.75ms$, the STAs showed biphasic shape without eventually getting flattened (figure 2.7 **b**).

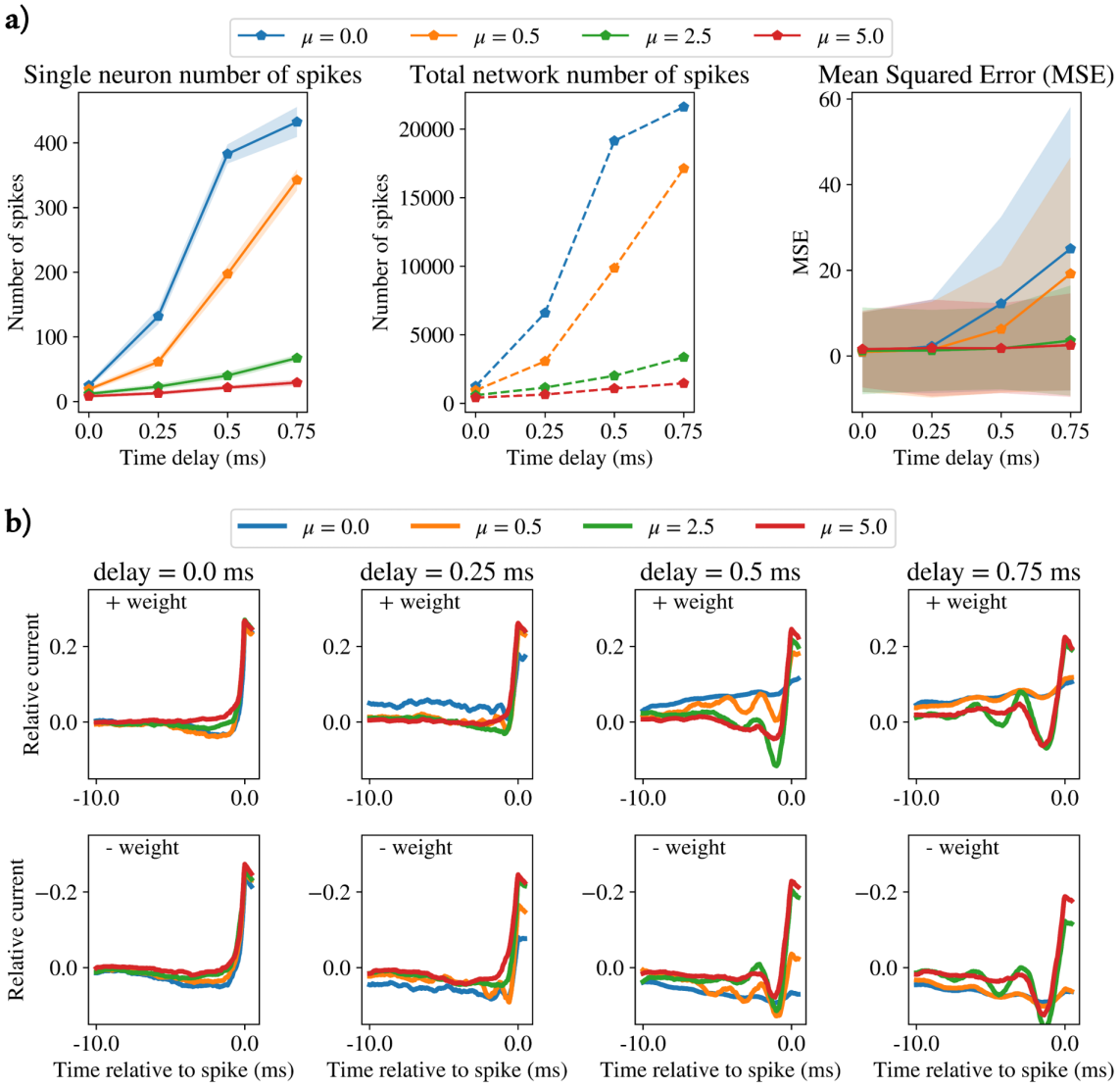

Figure 2.7: Interaction between time delays and cost term (μ). **a**) Left: The average number of spikes for a single neuron, the shaded areas represent the standard deviations. Center: Total number of spikes of the network. Right: MSE between the target output $\mathbf{x}(t)$ and estimate $\hat{\mathbf{x}}(\mathbf{t}$, the shaded areas represent the standard deviation. Each color represents a different cost term value, which also applies for panel **b**. **b**) STA filters with different time delays and cost term parameters. Every column represents a different time delay. In the top row, the STAs of all positively weighted neurons are plotted, and the neurons with negative weight are plotted in the bottom row. In the bottom row, the y axis of each plot is shifted. The size of the network remained constant with 50 neurons throughout the simulations to obtain the shown results in this figure.

2.4 Discussion

Theoretically derived network models are a common approach to studying information representation in the brain. Such models can implement relatable computations from the brain employing coupled LIF neurons (for example, [14, 15]). In this work, we focused on the *Poisson* BSN framework model, which can implement either dynamical systems or autoencoder computations with the presence of connectivity delays and non-deterministic neuron thresholds [15]. The *Poisson* BSN model shows bio-like spiking properties, is robust to perturbations, and has very efficient spiking. However, no attention has been paid about how the coding properties of neurons arise and are affected by neuron interaction in these network models. To investigate this, we implemented the *Poisson* BSN model, and the same STA analysis explained in the first chapter was applied to the modeled neurons from the network. By systematically changing the parameters that might affect the neurons' coding properties due to neural interaction (network size, spiking cost, and time delays), we investigated how the stimulus-response relationship of these neurons arose and how it affected the properties of the network.

Network size, spiking cost, and time delays affected the stimulus-response relationship of the neurons from the network and modulated network performance. These changes were not only seen when independently modifying one parameter but also when they were interacting. The stimulus-response relationship of the neurons based on the STA analysis was equivalent to type 1 and type 2 excitability, which changed based on the network parameter modulation.

We showed how an increase in network size improved network performance by reducing the error between the estimate $\mathbf{x}(t)$ and target output $\hat{\mathbf{x}}(t)$. So with more neurons in the network, more spikes are available to approximate the target output. This means more efficient spiking in single neurons but a higher total network spike count. Spiking became less efficient and more prone to error when the network size was around 400 neurons, which can be explained by the ping-pong effect [14]. This effect was caused because every time a spike occurred, the subsequent change in the estimate directly reached the error threshold of another neuron that would spike right after (for a geometrical perspective of the ping-pong effect, see [35]). Variations in network size also resulted in changes in the STA filters. In small networks, neurons might have been more sensitive to input intensity, showing a positive monophasic STA (type 1 excitability). On the other hand, larger networks' neurons could have been more sensitive to input frequencies rather than intensities (type 2 excitability). This might indicate changes in how information is represented [8]. Once the network had a ping-pong spiking behaviour, the STAs stopped being sensitive to specific input features related to the spike.

The cost term penalized a neuron for repetitive firing by extending its refractory period. Although it increased efficiency (fewer spikes in single neurons and at a network level), accuracy was undermined if the cost term was too high (≥ 50). In that situation, the neurons were not allowed to spike, although there was enough network error to spike. In a network where the baseline STA is biphasic, increasing the spiking cost of the neurons caused a shift to a monophasic STA. Due to this impediment, neurons could not be sensitive to frequency if they could not spike in precise timing, which led to a monophasic STA, where neurons would only overcome this cost if the error became high enough.

Delays in the recurrent connectivities caused a decrease in network performance: more error and more spiking in general (especially in larger network sizes). Small connectivity delays resulted in more synchronous firing patterns throughout the network (which was also previously seen in [36]). Longer delays ($< 5ms$) caused the ping-pong effect in the network (also seen in [35]). However, these effects contrast with results from the *Poisson* BSN implementation in [15], where the network showed more stable behaviour regardless of time delays (up to $\approx 2ms$ delays). A relevant factor that might have caused this is their use of higher alphas ($\alpha > 800$) compared to the values we used here ($\alpha = 10$).

Hence, higher alphas made spiking less probabilistic and possibly more resilient to delays (also in line with results found in [35]). In this network framework, neural synchronisation decreased network performance. This synchronization had a clear repercussion on the STAs of the neurons from the network since synchrony with type 2 excitability is easier to achieve than with type 1 neurons [37, 38]. However, the STA shape, could also have been explained because of cyclical spiking interfering with the actual neuron filter. For higher delays, such spiking synchrony faded due to strong ping-pong effects. These effects made the STA not sensitive to the input but rather, it was determined by the constant (delayed) updates the neuron was receiving.

The interaction between network size, cost term, and time delays had further effects on network performance and neural coding properties. As network size increased, modifying the cost term did not change network performance but changed the stimulus-response relationship of all neurons. This indicates that the input can be coded differently by the network and yet not interfere with network performance. This effect could be due to amplitude encoding at a network level since it has been observed that amplitude might not affect neural encoding performance regardless of whether neurons are type 1 or type 2 [39].

We saw a strong effect of time delays in network performance depending on the network size. Delays affected way more network performance for larger networks than for smaller networks. Since there are more spikes available in larger networks, a delayed update can also cause more erroneous spiking. In larger network sizes, delays generated a ping-pong effect which deteriorated the stimulus-response relationship of the neurons relevant to the stimulus. Small network sizes were especially resilient to time delays, so their STAs remained sensitive to the input features.

The ping-pong effect could be countered by the cost term, forcing the neurons to have lower firing rates. Interestingly, while a high cost term alone caused type 1 neural excitability, in presence of time delays, it caused type 2 excitability while avoiding the ping-pong effect. We saw that delays alone in the network had a synchrony effect on the spiking of neurons but eventually caused a ping-pong spiking effect. However, the spiking cost term combined with time delays prevented the neurons from spiking constantly and helped preserve neural synchrony and network efficiency.

In this research, we could see that by modifying network parameters from the *Poisson* BSN, we could change the stimulus-response relationship of the neurons given by the STA analysis, something that has previously been unexplored (except for [6], which used the BSN model). We showed that the stimulus-response relationship of the neurons is influenced and interacts at a network level, also relating to network performance. These changes might reveal important insights about how information propagates (in rate or spike code) in biological and modeled networks [40]. Indeed, the current research is still far from reproducing biologically realistic network behaviour, for instance, for its lack of neural heterogeneity. Therefore, a direct next step of this research is to make the *Poisson* BSN framework heterogeneous and study how the found input-output relationships arise in that context. To make the current framework heterogeneous, one could add different delays and spiking cost terms in the network's neurons. On top of that, a very promising approach would be to include neuron filters as done in [6]. These filters could differentiate the neurons' excitability between type 1 and type 2 in the network. By setting the neuron filters as done in [6] in the *Poisson* BSN network, one could study how the parameters can shift the pre-set neurons' coding properties, as the current research has shown.

Conclusions

How the brain carries and transforms information is still not fully understood. This problem ranges from single neurons to larger networks and brain areas. However, regardless of the brain scale, this issue can be investigated by studying how the input and the output relate. In this research, we highlighted the importance of understanding this relationship across scales to have a deeper understanding of the brain. We first investigated the effect of ACh on the coding properties of single neurons from experimental data using STA analysis. We found that it is likely that this neuromodulator can only shift from type 2 to type 1 excitability in a single neuron but not the other way around. After, we explored how neural input-output relationships arose in a network model by applying the same analysis used in the experimental data. We found that the coding properties of the modeled neurons are sensible to network parameters that could be relatable to what we can find in biological networks (such as different network sizes, spike cost terms, or connectivity delays).

The STA analysis allowed us to study how input and output relate to single neurons in experimental recordings and network modelling. However, this method has some limitations when it comes to unraveling the actual coding properties of neurons since it can be biased by input correlations or spike bursts. Although we should consider other measurements and analysis to reproduce and get closer to understanding the brain's coding properties, these biases can be corrected and be very useful to unravel some of the coding properties of neurons[27].

In Chapter 1, we found that all neurons had type 1 excitability regardless of the condition. This excitability type could be related to the neurons from small networks ($N = 2$) modelled in Chapter 2. However, it would be important to model the neurons from the analysed recordings in chapter 1 to be able to compare the STA filters between the recordings and the neurons from the *Poisson* BSN model. Also, how the experimentally observed neuron filters would affect network performance in the *Poisson* BSN model could be further explored by using a similar network derivation as done in [6]. In Chapter 2, we have shown changes in spiking generation dynamics due to network effects. These results bring new questions regarding the implications of recording single neurons in isolated settings, as we did in Chapter 1. Therefore, it could be that single cell in vitro-like conditions do not fully capture the network effects on single neurons [11].

In this thesis, we highlight the importance of bringing the same analysis to different brain scales to understand how coding arises in different contexts. In the past years, there have been increasing ways to measure entire biological neural networks at once [41]. Such data sets can be crucial in bridging the gap between mechanistic explanations of single neurons and network dynamics. And yet, analysing them with a similar approach as we did in this thesis will be very relevant to understanding such data sets. First, we can study single neuron dynamics and coding properties in a biological network context. And second, we will still need network modelling to understand the inner properties of these networks by linking them with what we observe in the data from single neurons recorded in a network context.

References

[1] Decharms, R. C. and Zador, A. "Neural representation and the cortical code". In: *Annual review of neuroscience* 23.1 (2000), pp. 613–647.

[2] Kriegeskorte, N. and Douglas, P. K. "Interpreting encoding and decoding models". In: *Current opinion in neurobiology* 55 (2019), pp. 167–179.

[3] Koch, C. and Laurent, G. "Complexity and the nervous system". In: *Science* 284.5411 (1999), pp. 96–98.

[4] Mejias, J. and Longtin, A. "Optimal heterogeneity for coding in spiking neural networks". In: *Physical Review Letters* 108.22 (2012), p. 228102.

[5] Perez-Nieves, N., Leung, V. C., Dragotti, P. L., and Goodman, D. F. "Neural heterogeneity promotes robust learning". In: *Nature communications* 12.1 (2021), pp. 1–9.

[6] Zeldenrust, F., Gutkin, B., and Denéve, S. "Efficient and robust coding in heterogeneous recurrent networks". In: *PLoS computational biology* 17.4 (2021), e1008673.

[7] Hodgkin, A. L. "The local electric changes associated with repetitive action in a non-medullated axon". In: *The Journal of physiology* 107.2 (1948), p. 165.

[8] Prescott, S. A. *Excitability: Types I, II, and III.* 2014.

[9] Izhikevich, E. M. *Dynamical systems in neuroscience*. MIT press, 2007.

[10] Prescott, S. A., De Koninck, Y., and Sejnowski, T. J. "Biophysical basis for three distinct dynamical mechanisms of action potential initiation". In: *PLoS computational biology* 4.10 (2008), e1000198.

[11] Prescott, S. A., Ratté, S., De Koninck, Y., and Sejnowski, T. J. "Pyramidal neurons switch from integrators in vitro to resonators under in vivo-like conditions". In: *Journal of neurophysiology* 100.6 (2008), pp. 3030–3042.

[12] Nadim, F. and Bucher, D. "Neuromodulation of neurons and synapses". In: *Current opinion in neurobiology* 29 (2014), pp. 48–56.

[13] Urai, A. E., Doiron, B., Leifer, A. M., and Churchland, A. K. "Large-scale neural recordings call for new insights to link brain and behavior". In: *Nature neuroscience* 25.1 (2022), pp. 11–19.

[14] Boerlin, M., Machens, C. K., and Denève, S. "Predictive coding of dynamical variables in balanced spiking networks". In: *PLoS computational biology* 9.11 (2013), e1003258.

[15] Rullán Buxó, C. E. and Pillow, J. W. "Poisson balanced spiking networks". In: *PLoS computational biology* 16.11 (2020), e1008261.

[16] Pang, R., Lansdell, B. J., and Fairhall, A. L. "Dimensionality reduction in neuroscience". In: *Current Biology* 26.14 (2016), R656–R660.

[17] Chichilnisky, E. "A simple white noise analysis of neuronal light responses". In: *Network: computation in neural systems* 12.2 (2001), p. 199.

[18] Pillow, J. *Likelihood-based approaches to modeling the neural code*. Vol. 70. 3. MIT press Cambridge, Massachusets, 2007.

[19] Stiefel, K. M., Gutkin, B. S., and Sejnowski, T. J. "Cholinergic neuromodulation changes phase response curve shape and type in cortical pyramidal neurons". In: *PloS one* 3.12 (2008), e3947.

[20] Roach, J. P., Eniwaye, B., Booth, V., Sander, L. M., and Zochowski, M. R. "Acetylcholine mediates dynamic switching between information coding schemes in neuronal networks". In: *Frontiers in systems neuroscience* 13 (2019), p. 64.

[21] Lucas-Meunier, E., Fossier, P., Baux, G., and Amar, M. "Cholinergic modulation of the cortical neuronal network". In: *Pflügers Archiv* 446.1 (2003), pp. 17–29.

[22] Metzner, W., Koch, C., Wessel, R., and Gabbiani, F. "Feature extraction by burst-like spike patterns in multiple sensory maps". In: *Journal of Neuroscience* 18.6 (1998), pp. 2283–2300.

[23] Gabbiani, F. and Koch, C. "Principles of spike train analysis". In: *Methods in neuronal modeling* 12.4 (1998), pp. 313–360.

[24] Yan, X., Calcini, N., Safavi, P., Ak, A., Kole, K., Zeldenrust, F., and Celikel, T. "A whole-cell recording database of neuromodulatory action in the adult neocortex". In: *bioRxiv* (2022).

[25] Zeldenrust, F., Knecht, S. de, Wadman, W. J., Denève, S., and Gutkin, B. "Estimating the information extracted by a single spiking neuron from a continuous input time series". In: *Frontiers in computational neuroscience* 11 (2017), p. 49.

[26] Arcas, B. A. y and Fairhall, A. L. "What causes a neuron to spike?" In: *Neural Computation* 15.8 (2003), pp. 1789–1807.

[27] Schwartz, O., Pillow, J. W., Rust, N. C., and Simoncelli, E. P. "Spike-triggered neural characterization". In: *Journal of vision* 6.4 (2006), pp. 13–13.

[28] Ermentrout, G. B., Galán, R. F., and Urban, N. N. "Relating neural dynamics to neural coding". In: *Physical review letters* 99.24 (2007), p. 248103.

[29] Mato, G. and Samengo, I. "Type I and type II neuron models are selectively driven by differential stimulus features". In: *Neural computation* 20.10 (2008), pp. 2418–2440.

[30] Barreiro, A. K., Thilo, E. L., and Shea-Brown, E. "A-current and type I/type II transition determine collective spiking from common input". In: *Journal of neurophysiology* 108.6 (2012), pp. 1631–1645.

[31] Stiefel, K. M., Gutkin, B. S., and Sejnowski, T. J. "The effects of cholinergic neuromodulation on neuronal phase-response curves of modeled cortical neurons". In: *Journal of computational neuroscience* 26.2 (2009), pp. 289–301.

[32] Buhl, E. H., Tamás, G., and Fisahn, A. "Cholinergic activation and tonic excitation induce persistent gamma oscillations in mouse somatosensory cortex in vitro". In: *The Journal of physiology* 513.1 (1998), pp. 117–126.

[33] Ermentrout, B., Pascal, M., and Gutkin, B. "The effects of spike frequency adaptation and negative feedback on the synchronization of neural oscillators". In: *Neural computation* 13.6 (2001), pp. 1285–1310.

[34] Paninski, L. "Convergence properties of some spike-triggered analysis techniques". In: *Advances in neural information processing systems* 15 (2002).

[35] Calaim, N., Dehmelt, F. A., Gonçalves, P. J., and Machens, C. K. "The geometry of robustness in spiking neural networks". In: *Elife* 11 (2022), e73276.

[36] Al-Darabsah, I., Chen, L., Nicola, W., and Campbell, S. A. "The Impact of Small Time Delays on the Onset of Oscillations and Synchrony in Brain Networks". In: *Frontiers in Systems Neuroscience* (2021), p. 58.

[37] Tateno, T., Harsch, A., and Robinson, H. "Threshold firing frequency–current relationships of neurons in rat somatosensory cortex: type 1 and type 2 dynamics". In: *Journal of neurophysiology* 92.4 (2004), pp. 2283–2294.

[38] Ermentrout, B. "Type I membranes, phase resetting curves, and synchrony". In: *Neural computation* 8.5 (1996), pp. 979–1001.

[39] Estarellas, C., Masoliver, M., Masoller, C., and Mirasso, C. R. "Characterizing signal encoding and transmission in class I and class II neurons via ordinal time-series analysis". In: *Chaos: An Interdisciplinary Journal of Nonlinear Science* 30.1 (2020), p. 013123.

[40] Kumar, A., Rotter, S., and Aertsen, A. "Spiking activity propagation in neuronal networks: reconciling different perspectives on neural coding". In: *Nature reviews neuroscience* 11.9 (2010), pp. 615–627.

[41] Pachitariu, M., Stringer, C., Dipoppa, M., Schröder, S., Rossi, L. F., Dalgleish, H., Carandini, M., and Harris, K. D. "Suite2p: beyond 10,000 neurons with standard two-photon microscopy". In: *BioRxiv* (2017), p. 061507.