Imperial College London

Department of Computing

Neural correlates of loss of consciousness in a spiking neuron model of the brain

authored by

Jaime Rodriguez

supervised by

Pedro A.M. Mediano

June 20, 2018

Abstract

Loss of consciousness (LoC) in the human brain occurs during deep non-rapid eye movement (NREM) sleep, general anesthesia and other altered states of mind. Multiple neural correlates of consciousness (NCC) have been suggested that highlight unique aspects of the dynamics of the brain and which can discriminate between different levels and contents of consciousness.

Whereas most existing research has focused on creating models that satisfy a single NCC, we have developed a detailed computational model of the thalamocortical system which satisfies several NCCs during LoC.

The model consists of 50,000 excitatory and inhibitory Izhikevich neurons organized in a modular structure according to human brain connectome data [18] and where each module oscillates within the Beta and Gamma frequency bands (15-100Hz). In an extension to this model, we added a thalamus-like structure which projects to the rest of the cortical model, motivated by the importance of this area of the brain in the homeostatic regulation of sleep.

Our model successfully reproduces the LoC dynamics as reflected by cortical synchronization [15] and the associated change in wave types from low-amplitude high-frequency waves to large-amplitude low-frequency delta waves. The thalamocortical model demonstrates the decrease in dynamic complexity (measured with the Lempel-Ziv complexity algorithm [25]) of its activity when the thalamus induces network-wide synchronization as the brain transitions from wakefulness to sleep.

Contents

Chapter 1

Introduction

1.1 Motivation

While previously challenged by most scientists, we now see many disciplines of science embracing the study of consciousness and attempting to break this mystical concept down into physical processes that we can rigorously study. Psychologists have contributed most of the existing research on theoretical frameworks to study consciousness, while neuroscientists have developed mathematical models of neurons and studied the dynamics of the brain in detail.

With the advances of modern day computers, we can take these mathematical models and theories of consciousness and bring them to life as computational models at a scale never imagined before. The transition from being passive observers measuring neural activity to constructing our own simulations based on experimental observation is of great importance. It allows us to test our knowledge by putting everything together in models based on our understanding of brain dynamics and connectivity and comparing the resulting dynamics with those of real brains. This results in an iterative cycle of improvement and refinement of our models and theories, and ultimately in pushing forward the boundaries of neuroscience.

The advancement of neuroscience is not only important to quench our curiosity about our inner workings and to find meaning to our own existence. It will also bring about clarity and perhaps remedies to neurological diseases and brain disorders that greatly affect our aging brains and bodies. In subtle ways, neuroscience plays vital roles in everyday medicine, for instance, in understanding the consciousness level of patients during surgery.

Our study is aimed at bridging the gap between our ability to measure levels of consciousness in patients through EEG, which has been proven possible [9], and our incomplete understanding of why and how changes in consciousness occur in the brain. The mismatch between our clinical capabilities and actual understanding of consciousness motivated our computational work of creating a model of the brain where we try to reproduce several neural correlates of consciousness in the brain.

1.2 Goals

The high level objective of this project has been to explore the available literature on neural correlates of consciousness (NCCs) and to extrapolate existing theories and measures into computational models that help us understand how loss of consciousness is reflected in the dynamics of the brain.

Our primary goal has thus been to construct a large-scale model of the brain with realistic human-like dynamics which reproduces several neural correlates of consciousness. Multiple such NCCs exist, some only theoretical and others more suitable for our use. We must do a literature review on existing NCCs, decide which we can measure and design a model that reproduces as many NCCs as possible.

To construct such a model, we must first provide a strong background on the construction of spiking neuron models, from low-level neuron models to large and complex simulations. Once our model(s) are completed, we should do a thorough analysis of their dynamics to demonstrate that the NCCs have indeed been satisfied.

1.3 Structure

This report is structured as follows. After this introduction, the Background chapter begins with Section 2.1, which presents the state of consciousness research, and Section 2.2, which reviews the available literature on NCCs. This is followed by an exploration of biological neuron simulation in Section 2.3, and an introduction to the `Brian2` framework for neuron simulation in Section 2.4, where we also introduce the neuron model implementations that we use throughout the experiments. To conclude the chapter, in Section 2.5, we give a brief overview of brain anatomy in relation to consciousness studies.

We then move on to the main body of the report, which starts off with an analysis of the methodology of our approach in Section 3.2. Section 3.3 contains the experiments that have been core to this report, which include: the introduction of the connectome in Section 3.3.1, the construction of a connectome network with individual neurons on every node in Section 3.3.2, the design of an oscillatory module of neurons in Section 3.3.3, the complete connectome experiment with oscillating modules in Section 3.3.4, an experiment with a thalamus-like central structure regulating the activity in Section 3.3.5 and the final experiment that aims to reproduce the PCI results in Section 3.3.6.

We close the report with our conclusions in Chapter 4.

Chapter 2

Background

Although our investigation is separate from the philosophical concept of consciousness, the term is related to our area of study and we find it of interest to briefly present the stance of research towards it. This is done in Section 2.1.

The neuronal correlates of consciousness (NCC) are defined as minimal sets of neuronal mechanisms that give rise to conscious experiences [11, 38, 33]. Although this definition seems to be linked to the philosophical idea of consciousness, the NCCs that we are interested in are *correlates of conscious content and level.* Thus, we can leave aside the troublesome and controversial study of the intangible concept of consciousness and focus on empirical measures that can be applied both to real and artificial brains. This chapter explores the different available NCCs in Section 2.2.

After this, we introduce the field of computational neuroscience, starting from the modelling of spiking neurons in Section 2.3. Here, we go over different existing models, each with their own advantages in terms of model accuracy and performance, but in the end, we choose one of them, in particular, Izhikevich Neuron models, for the bulk of our experimental work.

We continue in Section 2.4 by introducing the framework for neural simulation that we have chosen: `Brian2`. In this section, we present the models for individual neurons that will be the foundation of our simulations and show examples of how to create networks from these individual neurons.

Finally, as part of this brackground chapter, in Section 2.5 we will briefly discuss the anatomy of the human brain and how each area is thought to contribute to our conscious experiences.

2.1 A brief overview of consciousness research

As Chalmers described it in [10], the main problem that one faces when attempting to explain the properties of consciousness is how the experience of the redness of red or the painfulness in pain arises as a result of the brain's activity. This is commonly described as the *hard problem* of qualia, it is the fundamental question of how subjective experience arises. In contrast, we have the so-called *easy problem*, which is to explain how the brain gives rise to perception, cognition, learning and behaviour. The typical critique to the study of consciousness is that solving the *easy problem* will get us nowhere in solving the *hard problem*. Although nobody has proposed a solution to the *hard problem*, or how the *easy problem* can help solve it, we agree with the position of the neuroscientist Anil Seth, of focusing on what he calls the *real problem*, which deals with how to account for the properties of consciousness in terms of biological mechanisms. The stance of the *real problem* is that we shouldn't pretend that consciousness does not exist (as the *easy problem* does) but we should also not worry too much about explaining its existence [31].

In 1990, Crick and Koch introduced the concept of the neuronal correlates of consciousness [11], in an effort to create a framework that would allow the scientific study of consciousness. Later on, in 2003, together with Francis Koch, they proposed a neurobiological theory that attempted to explain the neural correlates of *visual* consciousness [12], thereby shifting their focus towards the correlates of conscious *content* (in this case visual content), and away from the more conteroversial and general idea of consciousness. Then, in 2008, Tononi and Koch proposed that the NCC should be investigated from two perspectives: firstly, studying neural dynamics when the *level* of consciousness is greatly affected, such as in the different states of sleep, during comma, vegetative states and seizures or during general anaesthesia; and secondly, analysing the *contents* of consciousness by investigating brain activity in relation to a specific type of conscious percept, such as visual consciousness [38].

This is the key point that we want to take away with us: when we refer to the NCCs throughout our study, we only care about the correlates of the *content and level of consciousness.*

2.2 Neural Correlates of Consciousness

As discussed in the previous section, our interest lies on the changes in content and level of consciousness. We can therefore study NCCs that can discriminate between changes in these.

We begin with the exploration of some well established NCCs and the motivation behind them. The measures presented include the integrated information measure Φ, causal density, Lempel-Ziv complexity and PCI.

In addition, we discuss a few other measures that can find characterizations of the level of consciousness in patterns of EEG activity, such as power and synchronization.

2.2.1 Integrated Information Theory (IIT)

The hypothesis presented in 2004 by Tononi states that consciousness corresponds to the ability of a system to integrate information. This is rooted in two key phenomenological properties of consciousness: differentiation, the fact that many different conscious experiences could potentially exist and the one we actually experience is one among all of these; and integration, the unity of this experience, the fact that our conscious experience is a whole, arising from the causal dependencies of all the individual components in our brain [37].

IIT attempts to combine these two properties in a single measure Φ, which is calculated mathematically through some measure of entropy across all possible bipartitions of the network (see the original IIT paper [37] for more details on the mathematics to calculate Φ). The number of bipartitions in a network has a factorial growth as the network size increases, therefore it is not feasible to compute in general. In addition, the IIT measure Φ has not been proved to be a good correlate of LoC. For these reasons, we will not apply IIT to our computational models.

Despite its lack of success as a practical NCC, the idea behind IIT has become the foundation for many other NCCs, such as PCI, which attempt to measure the amount of information and integration in a system, though in different ways.

2.2.2 Causal Density

Another existing NCC is causal density, which measures the fraction of neuronal interactions that are causally significant. It is typically measured through Granger causality, which is based on linear regression modelling of stochastic processes. It requires the modelling of each neural process as a stochastic process, at which point covariance matrices between them can be calculated to obtain a measure of causality.

A practical limitation linked to this complexity measure is that accurate estimation of these multivariate stochastic models becomes difficult as the number of variables grows. For multivariate regression models, the number of variables increases quadratically with the number of elements in the network [33]. Therefore, we will not be calculating this NCC for our model.

2.2.3 Lempel-Ziv Complexity

Another way to gauge the amount of information encoded by a signal is to measure its algorithmic complexity. Very much related to the Lempel-Zip-Welch (LZW) compression algorithm, is a measure of algorithmic complexity known as the Lempel-Ziv (LZ) complexity, introduced by Lempel and Ziv in their 1976 paper *On the complexity of finite sequences* [25]. This measure is introduced as a method of evaluating the complexity or randomness of finite sequences by an account of the number of steps required to generate this sequence, but is also related to the number of different substrings in this sequence and the rate of their occurence.

An increasing interest in dynamical complexity measures for brain activity has made LZ complexity a useful measure. LZ complexity has been used in PCI to quantify the complexity of spatiotemporal data (see Section 2.2.4). It has also been the complexity measure of choice as an entropy rate estimate of spike trains in [3]. Furthermore, we find evidence of the validity of LZ complexity as an NCC in the research by Schartner et al., where LZ complexity was employed to robustly measure a decrease in the complexity of spontaneous EEG activity during general anaesthesia [29]. Further research that uses LZ complexity on brain dynamics exists, see [1, 16].

The question remains of how to apply this measure to a whole computational simulation. We present this in the relevant section where it will be used, Section 3.3.5. The key idea is that, as professed by IIT, we want to measure the amount of information in the neural activity of the simulation, as well as the integration between distant areas in the brain. Thus, we must first record the activity of each module and then combine the data from all modules to obtain a final complexity measure.

2.2.4 Perturbational Complexity Index

In 2013, Casali et al. developed an NCC based on the ideas of integration and information that are core to IIT, which they named the Perturbational Complexity Index (PCI). PCI is computed by first stimulating the cortex with transcranial magnetic stimulation (TMS) to engage distributed interactions between multiple brain regions (integration) and then compressing the spatiotemporal pattern of resulting brain activity to obtain a measure of algorithmic complexity (information) [9]. A spatiotemporal matrix of significant activation sources is calculated, on which LZ complexity is computed and then normalized to finally obtain PCI.

PCI effectively measures how much (information) and how distributed (integration) a response to TMS in the brain is. With this measure, the authors were able to reliably discriminate the *level* of consciousness in individuals undergoing different states of wakefulness, sleep, anesthesia, as well as minimally conscious patients who had emerged from a coma[9]. This, of course, has great medical potential for identifying the level of consciousness of a patient without requiring them to perform any kind of sensory or cognitive task, which is important for states such as the locked-in syndrome where patients are fully conscious but unable to move or communicate verbally.

We provide technical details about how PCI is calculated in Section 3.3.6.

2.2.5 NCCs of sleep

Sleep is an interesting neurological process because it is the most evident example of loss of consciousness. We have all experienced the change in consciousness that our brain undergoes as we drift into deep sleep. There are different stages of sleep and in fact not all of them are considered to be a state of LoC. For instance, rapid-eye movement (REM) sleep is the sleep stage where we dream, and it turns out that our mind has a level of consciousness similar to that of wakefulness, though sensory input is “turned off”. Deep Non-REM sleep is the epitome of LoC, where our brain waves drastically change and consciousness is thought to disappear. Aside from natural sleep, LoC can be induced with pharmacological agents, such as halothane, propofol or benzodiazepines, which are employed during general anaesthesia.

Different *levels* of consciousness (wakefulness, REM sleep, deep dreamless NREM sleep, general anaesthesia, etc) come accompanied with very distinct and characteristic patterns of EEG activity. Measures that characterize these patterns can therefore serve as NCCs, much like PCI does.

Power

Thalamocortical oscillations markedly change in the sleeping and aroused brain, as Steriade et al. presented in [35]. When the brain falls asleep, the rapid patterns associated with wakefulness are replaced by low-frequency, synchronized rythms of neuronal activity.

The thalamocortical system during NREM sleep behaves as follows:

- At the onset of sleep, spindle oscillations at 12-15Hz are predominant.
- As sleep deepens, delta waves at 1-4Hz predominate and slow oscillations at 0.5-1Hz appear (also known as Slow-Wave Acivity, SWA).

The other type of sleep, associated with rapid eye movements (REM sleep) and dreaming, is characterized by fast oscillations, much like in wakefulness.

Upon arousal from deep sleep, the low-frequency oscillations of sleep are replaced by fast oscillations in the range of 20-80Hz (typically around 40Hz). Furthermore, EEG recordings show a transition from low-amplitude, high-frequency rythms, to large-amplitude, low-frequency oscillations [35].

Synchronization

An interesting result, again related to the neurodynamical properties of the brain during arousal and sleep, is that of cortical synchronization.

A study by Esser et al. revealed the role of cortico-cortical synaptic strength in regulating network synchronization in the cortex. Their findings show that decreased synaptic strength led to reduced network synchronization and emergence of local clusters of synchronized activity [15]. This resulted in a reduction of SWA and marked the transition from sleep to wakefulness.

These characteristics of sleep dynamics (delta power and cortical synchronization) thus serve as NCCs, which we will apply in Section 3.3.4.

2.3 Computational neuron simulation

Leaving behind the NCCs and consciousness research, we now dive into the field of computational neuroscience.

Typical neurons are made up of three functional parts: dendrites, which collect electrical signals from neighbour neurons and pass it to the soma; the soma, or body of the neuron, which is responsible for a non-linear processing of the input signals and may generate an output electrical signal and emit it through the axon; and the axon, which transports electrical signals to the dendrites of other neurons.

The electrochemical junction between two neurons is called the synapse. This gives way to referring to the neuron that sends a signal as the pre-synaptic neuron and the one receiving it as the post-synaptic neuron. In addition, the electrical pulses of these cells are typically referred to as action potentials or spikes. By measuring the voltage difference across the cell membrane of the neuron, typically called the membrane potential, we can record the spiking behaviour of neurons.

As spikes from pre-synaptic neurons arrive at a post-synaptic neuron, the voltage of the cell increases as a summation process (also referred to as integration) of all the incoming signals. However, once the membrane potential reaches a certain threshold θ, the cell undergoes a strong depolarization phase (where the voltage suddenly increases) followed by a repolarization phase where the potential goes back to its resting value. We can see this effect in Figure 2.1.

Since action potentials typically have a very similar shape, it is not the exact shape of the spikes that represents an exchange of information between neurons, but rather it is their presence or absence which represents information. In fact, the frequency and timing of these spikes is what influences the dynamics of a network of neurons.

We now present three different spiking neuron models. The first one, the Leaky Integrate and Fire model, is the simplest neuron model, an approximation that does not accurately reproduce all aspects of a neuron's dynamics but may be of use for some applications. Secondly, we introduce the most

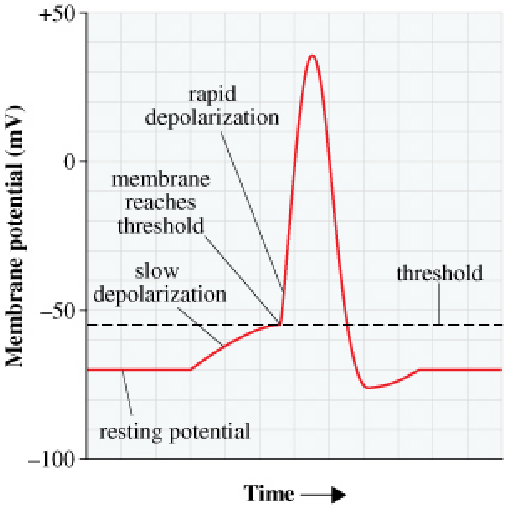


Figure 2.1: Neuron action potential

complete and biologically-accurate model known, the Hodgkin-Huxley model, which models low-level aspects of neurons' dynamics, such as the conductance of different ions across the cell membrane, and hence faithfully replicates all aspects of their dynamics, but it is more expensive to simulate than the simple LIF model. Finally, we discuss the Izhikevich neuron model, which was developed as a mathematical model that replicates the dynamics of a wide range of spiking neurons in the brain with great accuracy, yet obviates the need to worry about low-level details such as ion channels and conductances. This middle ground is perfect for our purposes, as it offers better performance than the Hodgkin-Huxley model and somewhat more realistic dynamics than the LIF model.

In the following sections we explain these three models in detail, because as a whole, they help depict the spectrum of existing biological neuron models, which are the foundation of the field of computational neuroscience. However, after this section, and for the remainder of the report, we will only use the Izhikevich model.

2.3.1 Neuron Model 1: Leaky Integrate-and-Fire Model

The first type of neuron model typically studied is the *leaky integrate and fire* model (LIF), which describes action potentials as events. This model represents the cell membrane as a basic RC-circuit, with the main equation for the model being an ordinary differential equation:

$$\tau_m \frac{du}{dt} = [u(t) - u_{rest}] + RI(t) \tag{2.1}$$

where τ_m is a time constant of the membrane, $u(t)$ is the membrane potential, u_{rest} is the membrane potential at rest (a constant), R is the resistance of the membrane and $I(t)$ is the driving current through the membrane.

In the absence of an input current, i.e. $I(t) = 0$, and assuming that at time t_0, $u(t_0) = u_{rest} + \Delta u$, the solution to Equation 2.1 is

$$u(t) - u_{rest} = \Delta u \, \exp\left(-\frac{t - t_0}{\tau_m}\right), \qquad \text{for } t > t_0 \tag{2.2}$$

where we see that $u(t)$ decreases exponentially to its resting potential.

If we take a constant input current, $I(t) = I_0$ which starts at $t = 0$ and finishes at $t = t_0$, then our ODE has a different solution:

$$u(t) = u_{rest} + RI_0 \left[1 - \exp\left(-\frac{t}{\tau_m}\right)\right], \qquad \text{for } 0 < t < t_0 \tag{2.3}$$

We can see the combined behaviour of Equation 2.2 and Equation 2.3 in Figure 2.2, where a short input current is followed by the absence of current.

The LIF model defines the firing time $t^{(f)}$ as the time when the membrane potential reaches the threshold voltage ϑ, i.e. $u(t^{(f)}) = \vartheta$. After the neuron fires, we reset the voltage to the resting potential. In Figure 2.3 we observe a spike train (a sequence of spikes) for a constant input current and a firing threshold ϑ. The upward arrows indicate the times when the neuron spikes. For a time-dependent input current, a more complex solution exists, which is discussed in [17].

This basic model does not have a refractory period, which is the period after a spike during which a real neuron is unable to fire again even if it receives a high dentritic current. A solution is to introduce an absolute refractory period to the equations governing the LIF model, during which the neuron is not allowed to spike. In addition, there are other IF models, such as the Quadratic Integrate and Fire model, which has a quadratic ODE, which are more realistic. In general, the IF model is computationally inexpensive, but it neglects many aspects of the real dynamics of neurons. We now discuss a model that is a lot more accurate biologically.

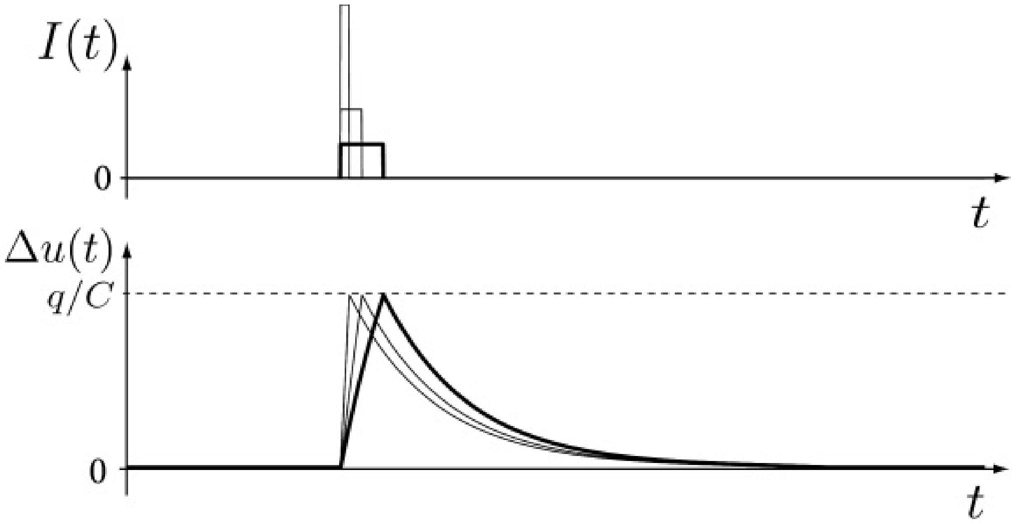


Figure 2.2: Constant input current in LIF model [17]

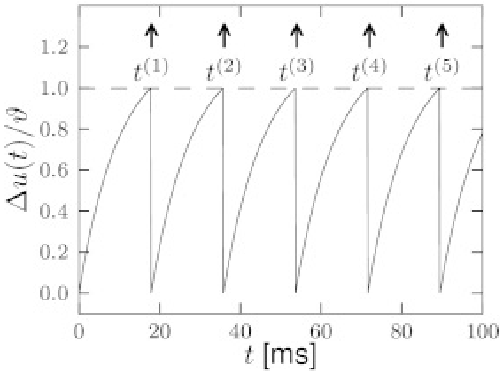


Figure 2.3: Spike train of a LIF neuron displaying the voltage normalized with ϑ for a constant input current [17]

2.3.2 Neuron Model 2: Hodgkin-Huxley model

By thoroughly studying the biophysics of neuronal dynamics, Hodgkin and Huxley created a more accurate model of the neuron membrane based on the flow of sodium (Na^+) and potassium (K^+) ions across the cell membrane. Once again, the model was that of an RC-circuit. Given an input current $I(t)$, this current charges a capacitor C, representing the membrane, but also leaks through sodium and potassium channels as well as other unspecified leaks. The leaks are modelled as resistances in parallel. The model of the membrane and its RC-model representation is shown in Figure 2.4.

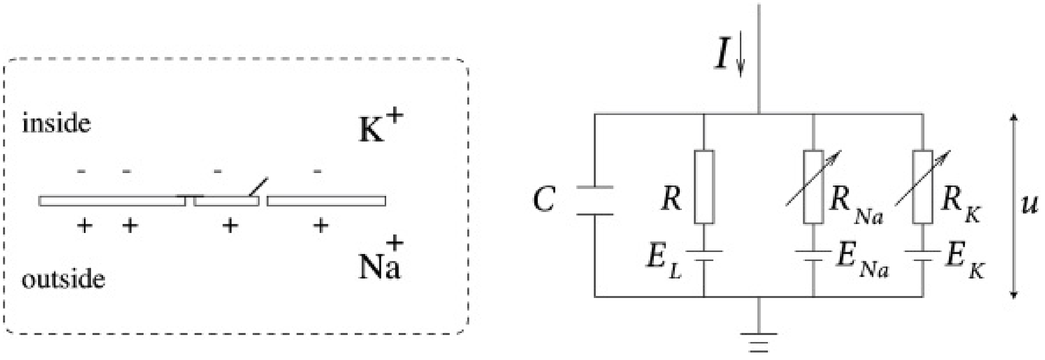


Figure 2.4: Hodgkin-Huxley model of the ion channels and RC-circuit neuron model [17]

In the circuit, E_K and E_{Na} are the potassium and sodium reversal potentials, related to the ion concentration of potassium K^+ and sodium Na^+ ions on either side of the cell membrane. We will now describe the solution to this circuit.

The capacitor charges following the equation

$$I_C = C\frac{du}{dt} \tag{2.4}$$

We also have that $I = I_C + I_L + I_{Na} + I_K$, the currents going through each of the parallel branches of the circuit. Given Ohm's Law, the current flowing through the leak resistor and voltage source is given by:

$$I_L = g_L(u - E_L) \tag{2.5}$$

where $g_L = 1/R_L$ is the conductance of the resistor. Similar equations apply for the Na^+ and K^+ channels.

This leads to the differential equation

$$C\frac{du}{dt} = g_{Na}(E_{Na} - u) + g_K(E_K - u) + g_L(E_L - u) + I \tag{2.6}$$

where g_{Na} and g_K are voltage-dependent conductances and the leak conductance g_L is constant. With a careful analysis of the sodium and potassium channels at the subcellular level, Hodgkin and Huxley further proposed models for these voltage-dependent conductances, to produce the complete Hodgkin-Huxley model:

$$C\frac{du}{dt} = \overline{g_{Na}}m^3h(E_{Na} - u) + \overline{g_K}n^4(E_K - u) + g_L(E_L - u) + I \tag{2.7}$$

$$\frac{dx}{dt} = -\frac{1}{\tau_x(u)}[x - x_0(u)] \tag{2.8}$$

where x stands for the gating variables m, h and n and $\overline{g_{Na}}$ and $\overline{g_K}$ are the maximum conductances of the sodium and potassium channels respectively. The gating variables m and h relate to the probability of the sodium channels to be open, n relates to that of the potassium channels and m_0, h_0 and n_0 are variables related to the opening and closing rate of these channels. The interpretation for the differential equation given by x governing m, h and n is straightforward: for a fixed voltage u, the variable x approaches the target value $x_0(u)$ with a time constant $\tau_x(u)$[17].

Parameters for this model have been fitted for different types of cells. For instance, Wulfram Gerstner summarises in Chapter 2 of Neuronal Dynamics [17] the fitted parameters for pyramidal neurons in the cortex as found by different researchers. Figure 2.5 shows an action potential of a Hodgkin-Huxley neuron, where we can see the dynamics of this model in more detail. After t = 2ms, we see the action potential followed by a relative refractory period where the potential of the neuron is below the resting potential. We can see a spike train for a Hodgkin-Huxley neuron with constant input current in Figure 2.6.

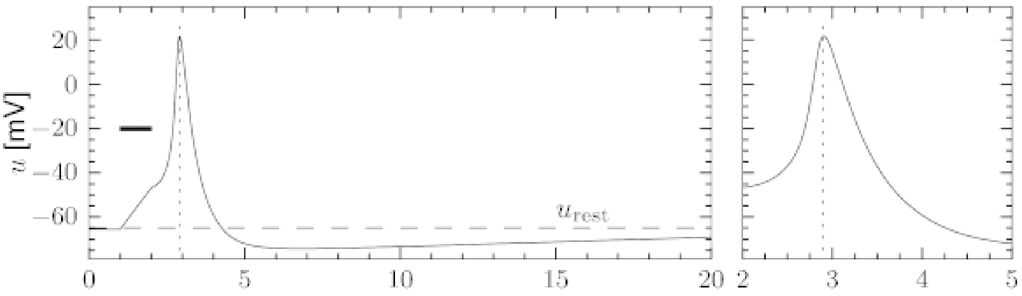


Figure 2.5: Action potential of a Hodgkin-Huxley neuron when stimulated by a short but strong current between t = 1 and 2 ms. Zoomed in image on the right.

This model is more accurate in reproducing biological features of neurons, such as the refractoriness, than the leaky IF model.

As we can see from Equation 2.7 and Equation 2.8, it is a system of 4 differential equations, which is generally difficult to analyze due to its relatively

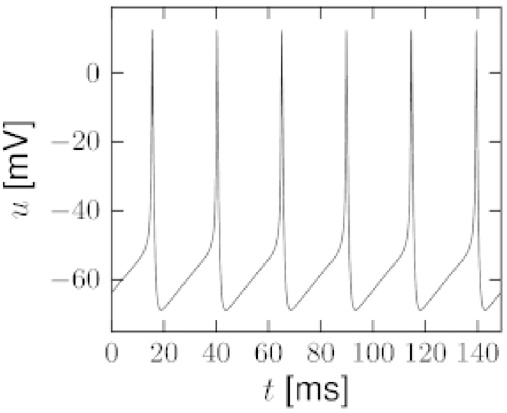


Figure 2.6: Spike train of a Hodgkin-Huxley neuron with a constant input current

high computational cost. For this reason, other models exist that reduce this model to two dimensions, such as the Fitzhugh-Nagumo and the Morris-Lecar model.

2.3.3 Neuron Model 3: Izhikevich model

Izhikevich neurons were proposed in 2003 by Eugene Izhikevich as a good compromise between computational efficiency (much better than Hodgkin-Huxley) and a biologically realistic repertoire of behaviours [22].

The proposed model is the following:

$$\frac{dv}{dt} = 0.04v^2 + 5v + 140 - u + I \tag{2.9}$$

$$\frac{du}{dt} = a(bv - u) \tag{2.10}$$

with the auxiliary after-spike resetting

$$\text{if } v \geq 30 \text{ mV, then } \begin{cases} v \leftarrow c \\ u \leftarrow u + d \end{cases} \tag{2.11}$$

Depending on the parameter combination of a, b, c and d, we can get different behaviours. For instance, we can build regular-spiking (excitatory) neurons or fast-spiking (inhibitory) neurons by changing a, b, c and d. In fact, in the original paper where the model was proposed [22], Izhikevich provided fitted values of the parameters to simulate all known types of neurons.

2.3.4 TMS modelling

Transcranial magnetic stimulation (TMS) is a technique to activate or deactivate specific areas of the brain's cortex in a non-invasive, non-painful manner [19].

Esser et al. constructed a computational model of the thalamocortical system and of TMS where they were able to reproduce the effects of TMS on the brain [14]. They modelled TMS as a synchronous activation of a chosen fraction of fiber terminals throughout the motor cortex. For a single cell, a single TMS pulse was modelled as transiently changing the conductance for a specified fraction of the cells synaptic channels.

An exact replica of their approach can only be done through a Hodgkin-Huxley-type model, whereas we will be dealing with Izhikevich neurons for the remainder of the project. Thus, for the PCI experiment in Section 3.3.6, we will model TMS as a synchronous activation of some fraction of the neurons in the model that are located spatially nearby.

2.4 `Brian2` framework

For this project, I have chosen the `Brian2` software platform for building the computational models of biologically-plausible neural networks. `Brian2` is an open-source[1] Python implementation of a spiking neuron simulator.

We will now run through an example on how to simulate a simple network of unconnected Izhikevich neurons using `Brian2`.

```python
# Define all our variables
Vt=30*mV
a=0.02/ms
b=0.2/ms
c=-65*mV
d=8*mV/ms
neuron_eqs = """
  dv/dt=(0.04/(mV*ms))*(v**2)+(5/ms)*v+140*mV/ms-u+I : volt
```

[1] https://github.com/brian-team/brian2

```
  du/dt=a*(b*v-u) : volt/second
  I : volt/second
"""
reset_eq = """
  v=c
  u=u+d
"""
thresh_eq = "v>=Vt"

# Create a neuron group for the simulation
# We will use 'rk4', i.e. Runge-Kutta 4th order numerical
# integration method to solve our model equations
G = NeuronGroup(100,
                neuron_eqs,
                threshold=thresh_eq,
                reset=reset_eq,
                method='rk4')
# Define the input current
G.I=4*mV/ms
# Simulation time
duration=1000*ms

# Initialise the parameters of the neurons randomly
G.v='(100*rand()-65)*mV'
G.u='-10*rand()*mV/ms'

# Create two Monitors to record the spikes of all the neurons
# and the exact voltage evolution of only the first neuron.
statemon_v = StateMonitor(G, 'v', record=0)
spikemon = SpikeMonitor(G)

# Run the simulation
run(duration)

# Plot the spikes
figure(figsize=(12,4))
subplot(121)
plot(spikemon.t/ms, spikemon.i, '.k')
# Plot the voltage of the first neuron
subplot(122)
plot(statemon_v.t/ms, statemon_v.v[0])
```

Code Listing 2.1: Izhikevich neurons in `Brian2`.

The above code produced the plots in Figure 2.7. See how the resulting dynamics are very similar to the Hodgkin-Huxley spike train shown in Figure 2.6.

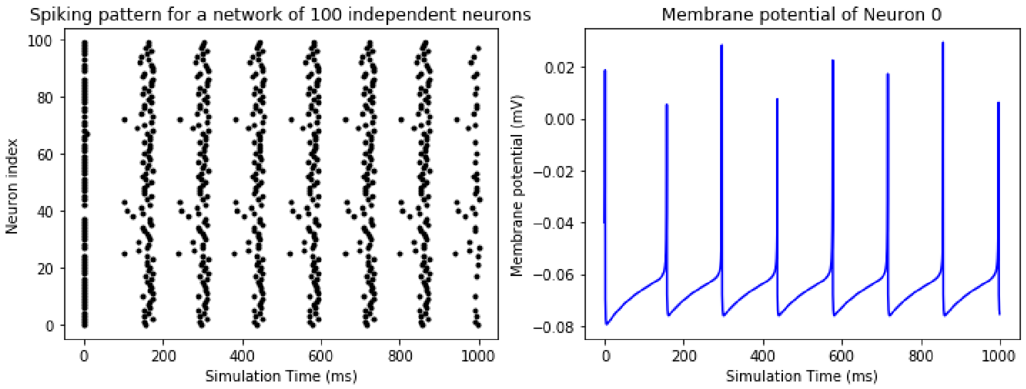


Figure 2.7: Izhikevich neurons simulated in `Brian2`. The black dots in the figure on the left-hand side represent spikes. The right plot shows the voltage over time of a single neuron from the simulation.

2.4.1 Excitatory and inhibitory neurons

Neocortical neurons in mammals display a range of spiking and bursting patterns, but they can be widely classified into excitatory and inhibitory neurons. As Izhikevich presented in *Simple Model of Spiking Neurons*, excitatory neurons may perform regular spiking, intrinsic bursting or chattering; and inhibitory neurons may undergo fast spiking or low-threshold spiking. All these behaviours can be displayed with the Izhikevich model by fitting specific parameters for the model [22]. In Listing 2.3, we show the common setup for all izhikevich neurons, both excitatory and inhibitory.

We now show how we modelled our Izhikevich neurons using `Brian2`.

Firstly, define the Izhikevich neuron model:

```
from brian2 import *

izhikevich_equations = '''
  dv / dt = (0.04 / (mV * ms)) * (v ** 2) + (5 / ms) * v +
                                  140 * mV /ms - u + I : volt
  du / dt = a * (b * v - u) : volt / second
  I : volt / second
  rnd : 1
  a : 1 / second
  b : 1 / second
  c : volt
  d : volt / second
'''

```

```
reset_equations = '''
  v=c
  u=u+d
'''

Vt = 30 * mV
threshold_equation = 'v >= Vt'
```

Code Listing 2.2: Izhikevich neuron equations in `Brian2`.

In Listing 2.2 we describe the differential equations corresponding to Equations 2.8 and 2.9 and declare the types of the parameters of the model, together with their units (which must be correct in order to be a valid Brian2 equation).

In *Simple Model of Spiking Neurons*, Izhikevich describes the best way to achieve heterogeneity so that different neurons have different dynamics, as is the case in the brain. **Excitatory neurons** should be assigned:

$$(a_i, b_i) = (0.02, 0.2) \tag{2.12}$$

$$(c_i, d_i) = (-65, 8) + (15, -6) * r_i^2 \tag{2.13}$$

where r_i is a random variable distributed over the unit interval and i is the neuron index. The random variable r_i marks the spectrum of dynamics from the regular spiking neuron through to the intrinsically bursting and the chattering neurons. Similarly, **inhibitory neurons** are assigned

$$(a_i, b_i) = (0.02, 0.25) + (0.08, -0.05) * r_i \tag{2.14}$$

$$(c_i, d_i) = (-65, 2) \tag{2.15}$$

to achieve the right heterogeneity. We use these parameters to create neuron groups in `Brian2` in the following code snippet.

```
def ExcitatoryNeuronGroup(N):
    G = NeuronGroup(N,
        izhikevich_equations,
        threshold=threshold_equation,
        reset=reset_equations,
        method='rk4'
    )
    G.rnd = 'rand()'
    G.a = 0.02/ms
    G.b = 0.2/ms
    G.c = '(-65 + 15 * (rnd ** 2)) * mV'
    G.d = '(8 - 6 * (rnd ** 2)) * mV / ms'
```

```
33      return G
34
35  def InhibitoryNeuronGroup(N):
36      G = NeuronGroup(N,
37          izhikevich_equations,
38          threshold=threshold_equation,
39          reset=reset_equations,
40          method='rk4'
41      )
42      G.rnd = 'rand()'
43      G.a = '(0.02 + 0.08 * rnd) / ms'
44      G.b = '(0.25 - 0.05 * rnd) / ms'
45      G.c = -65 * mV
46      G.d = 2 * mV / ms
47      return G
```

Code Listing 2.3: Excitatory and inhibitory neuron groups declared using `Python` and `Brian2`. The methods are parameterised with the variable N, representing the number of neurons to be declared.

One very important aspect about the above neuron group definitions is the fact that we have introduced a random variation into the parameters and hence into the dynamics of each neuron. `Brian2` allows to declare variables as expressions that are evaluated by the `Brian2` framework. This is possible for special objects, such as the `NeuronGroup` object. In Listing 2.3, lines 28 and 42 contain the declaration `G.rnd = 'rnd ()'`, that associates a string to a property of the `NeuronGroup` object. This string will be interpreted by `Brian2` for each of the N neurons in the `NeuronGroup`, hence associating a different random variable to each neuron.

What are the resulting neuron dynamics of these models? Figure 2.8 shows the dynamics of two excitatory neurons that are fed a constant current of 5 mV/ms. They differ in the random parameter r_i, which is used in Equations [2.13,2.15] to generate variability in the dynamics.

Similarly, Figure 2.9 shows the effect of the random variable on the neuron dynamics of inhibitory neurons. In this case, the main change is the inter-spike interval, which decreases with a larger r_i value. To more realistically represent the range of inhibitory neuron dynamics, more parameters of the Izhikevich model need to be varied than what Equation 2.15 provides, but this is a simple and efficient solution that satisfies our present requirements.

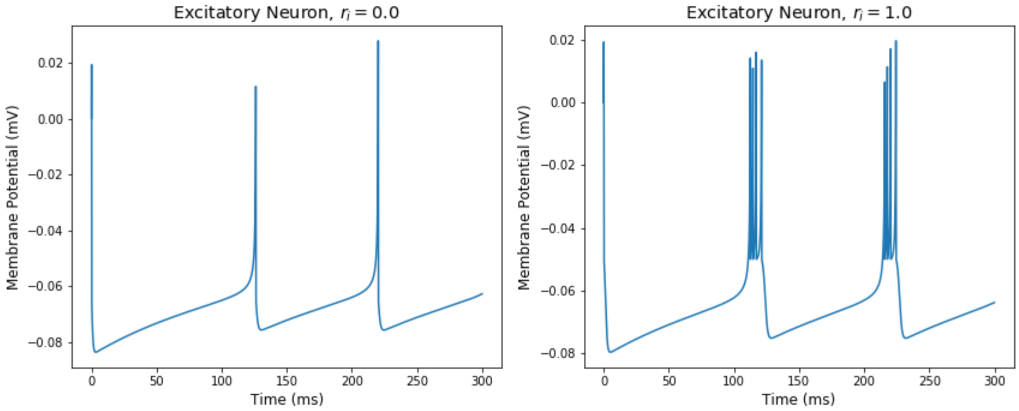


Figure 2.8: Excitatory neuron dynamics simulated with `Brian2` with extreme values for the random variable $r_i = [0.0, 1.0]$, displaying regular spiking behaviour (left) and chattering (right). Both neurons receive a constant input current of 5 mV/ms.

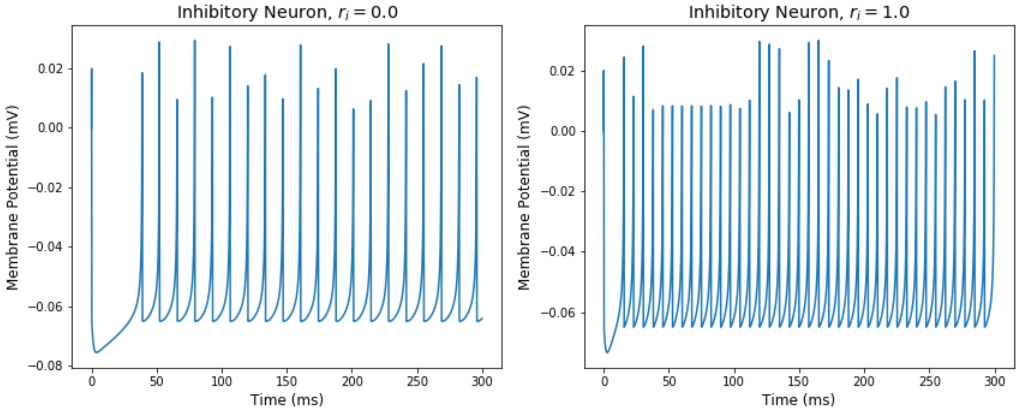


Figure 2.9: Inhibitory neuron dynamics simulated with `Brian2` with extreme values for the random variable $r_i = [0.0, 1.0]$, displaying fast spiking (left and right) with increased frequency in the right figure. The neurons receive a constant input current of 10 mV/ms.

2.4.2 Synapses

To set up a `Synapse` group, you provide the neuron groups it is connecting and an expression describing what happens when a presynaptic spike reaches the postsynaptic neuron. Then, the actual synapses between the individual neurons are specified by calling `.connect` on the `Synapse` object and providing a connection condition or the actual indices of the neurons that we want to connect, as seen in Listing 2.4. The method call to `.connect` is time consuming and must be called as few times as possible. In terms of performance, it is faster to directly specify all neuron indices involved in the synapses or use the string generator syntax to create all synapses at once rather than having to call `.connect` repeatedly with simple connect statements.

```
G1 = NeuronGroup(...)
G2 = NeuronGroup(...)
S = Synapses(G1, G2,
    model='w: volt',
    on_pre='v += w',
)
# Connect neurons from G1 to G2 that satisfy the condition.
# i and j are the indices of the pre-synaptic and
# post-synaptic neuron, respectively.
S.connect(condition='abs(i-j) < 5')
```

Code Listing 2.4: Example Synapse creation

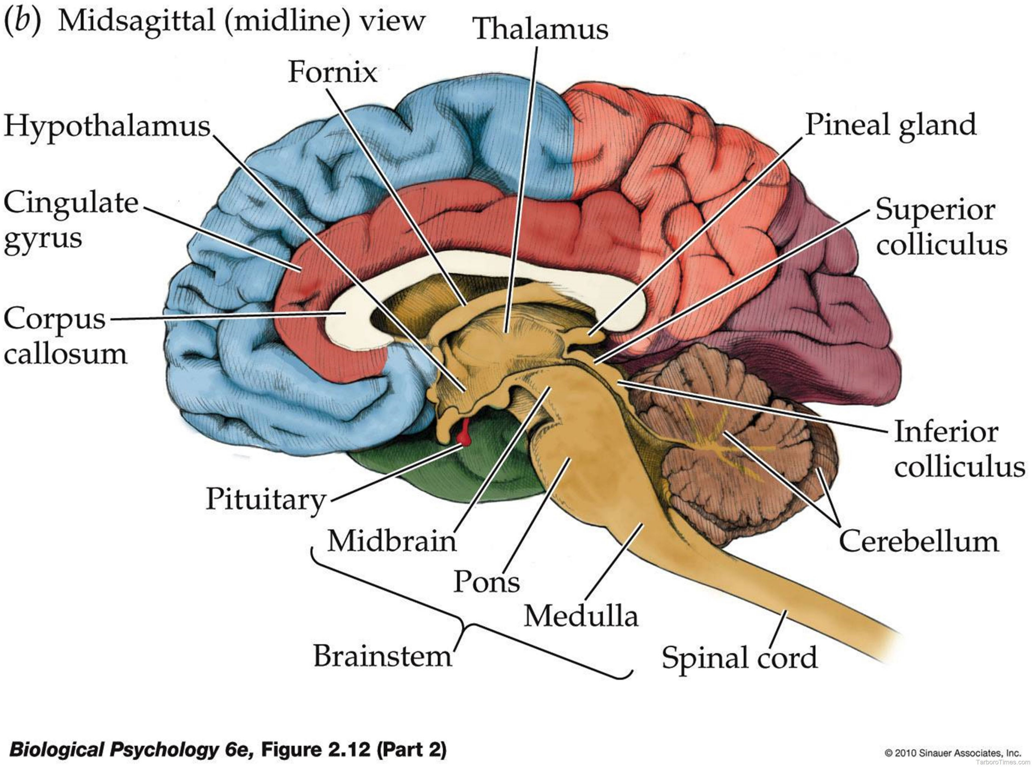
2.5 Brain anatomy

In order to simulate the brain and study the properties of conscious systems, we faced the challenge of modelling one of the most complex organs in our bodies. Before we delve into our computational models, it is worth investigating what we already know about the brain, and what is of interest to our study of consciousness and brain circuits.

The brain, together with the spinal cord, constitutes our central nervous system (CNS), but the nervous system also includes the peripheral nervous system (PNS), made up of a circuit of nerves extending to every corner of our body [36].

2.5.1 Cerebrum

The **cerebrum** is the largest and most recognizable part of the brain, consisting of two hemispheres, left and right, each of which is made up of an outer region of gray matter known as the cortex and an inner one, the white matter, which contains all the cell axons of the cortex neurons [26]. The **cortex**, with its distinctive folded and wrinkled structure, is a thin layer of


(*b*) Midsagittal (midline) view

***Biological Psychology 6e*, Figure 2.12 (Part 2)**

© 2010 Sinauer Associates, Inc.

Figure 2.10: Diagram of the human brain [28]

neurons covering the whole surface of the cerebrum; seen in Figure 2.10 as the regions painted in blue, red, orange, purple and green. It contains the cell bodies of all cortical neurons, the axons of which extend to form the white matter of the cerebrum. The cortex is typically divided into four regions or lobes: the frontal, parietal, occipital and temporal lobe, each associated with a wide range of functions, including reasoning, emotions, movement, visual processing, perception, memory, etc. This is the core structure that we want to study.

2.5.2 Cerebellum

The **cerebellum** is located at the back of the brain, behind the point where the spinal cord meets the brain. This dense region of neurons coordinates voluntary movements such as posture, balance, coordination and speech, providing balance to muscular actions and participates in motor learning. This structure is usually ignored in studies of consciousness.

2.5.3 Limbic system

A peculiar region of the brain is the **limbic system**, which includes a set of complex structures like the *thalamus*, *hypothalamus*, *amygdala* and *hippocampus*. This area is primarily responsible for our emotions and the formation of memories [6].

The **hippocampus** plays a crucial role in converting short-term memories into long-term ones. The **hypothalamus** drives the homeostasic regulation of processes such as hunger, thirst, response to pain, pleasure, anger, etc. To drive this regulatory response, the hypothalamus interacts with the autonomic nervous system and the pituitary gland, which releases hormones into our bloodstream. The **amygdala** is a small structure mainly related to agressive responses to stimuli [6].

The **thalamus** relays all the sensory information to the appropriate area of the cortex for correct processing of these signals. It acts as a central sorting station for sensory input, but also participates in a recurrent loop of activity between the cortex and itself. The cortex and the thalamus are repricocally connected, forming the thalamocortical loop. The cortex receives sensory stimuli through the thalamus, creates multiple conflicting hypothesis in response to the stimulus, which are then integrated again by the action of the thalamic neurons and once again sent back to the cortex [27]. The thalamus is thus crucial in processing and interpreting sensory input in our brain.

As it turns out, the thalamus also plays a central role in consciousness: when we sleep, the cortex becomes functionally disconnected from the sensory outside world because the thalamus blocks the transmission of any sensory information to it. This occurs during loss of consciousness, as happens during sleep as well as when induced by other anaesthetic pharmacological agents, e.g. halothane, propofol, benzodiazepines [2]. The block in sensory transmission occurs because of a hyperpolarization of the thalamocortical cells, which as a consequence change their predominant firing pattern. During LoC, thalamocortical neurons, which typically exhibit tonic firing, start firing in burst patterns. Alongside this change in dynamics, we see the EEG activity patterns change from the fast activity that marks wakefulness, to the slow waves representative of sleep [2, 34].

2.5.4 Summary

Other regions exist, such as the brainstem and spinal cord, but none are relevant to our study. The takeaway from this section is that the cortex and the thalamus, which together form the thalamocortical system, are the core structures of interest in consciousness studies, and are therefore the areas that we will focus on.

Chapter 3

Body of Report

3.1 Introduction

After revising the relevant theory for our work, we can move on to the experiments and their results. We will do our best to describe in detail the problems faced and the decisions we made at every step of the experimental process.

Section 3.2 opens this chapter with a review of the factors that have motivated the specific design decisions and trade-offs made in this project. Then, in Section 3.3, we dive right into the experiments that are the core of this report.

3.2 Methodology

Building a complete brain model that is suitable for our goals requires a process of iterative evaluation and redesign. We now summarize the different aspects that we have taken into account.

3.2.1 Simulation size and computational load

A typical brain has 100 billion neurons with an average neuron having 10^4 synaptic pathways to other neurons. This would result in a full model with

10^{15} synapses. The sheer size of these numbers makes any full-scale computational model of the brain, even if run on a current day multi-core supercomputer, a tremendously slow process. This is undesirable for multiple reasons: first of all, the elevated cost required to rent a cluster with enough cores to run this simulation; secondly, with such an expensive model, it would take days to run a single second of simulation, as Izhikevich showed in [21], where he constructed a real-scale model that took 50 days to run 1 second of simulation.

The insight we are seeking when simulating the brain does not come exclusively from having a very large simulation; instead it comes mostly from its qualitative likeness to the brain. It is in some way desirable, however, to create a large-enough model in order to obtain more complex, brain-like dynamics. Some reaserchers have been pushing these boundaries, creating models with millions of neurons and billions of synapses [23, 13]. Our complete model, as presented in Section 3.3.4, will consist of 50,000 excitatory and inhibitory Izhikevich neurons, arranged in oscillatory modules. This experiment size lies in the limit of computationally feasible simulations without the need of large computing clusters, while still allowing us to obtain a good amount of good quality results.

3.2.2 Heterogeneity

The network needs to incorporate heterogeneity into its structure and dynamics, otherwise it cannot hope to mimic the large behavioural spectrum present in biological neural networks. In Section 2.4.1 we describe how we have achieved heterogeneity of neuronal dynamics by sampling the parameters for our neurons from probability distributions. In addition, the final model incorporates variability in the oscillatory dynamics of the different modules, since the precision and frequency of oscillation is also probabilistically sampled.

3.2.3 Neuron model accuracy

Biological accuracy in a computational model depends on many factors. A few of these include: modelling the ion channels in the neuron model, simulating the cable equation for the transmission of electric impulses throughout the axon of a neuron, modelling the neuron as a multicompartmental object, the modelling of the synapse, plasticity, etc. There is, however, great benefit from simplifying the neuron and network models, mostly for performance and scalability purposes. We make use of the simpler and more efficient Izhikevich neuron models, that provide a great tradeoff of performance and realistic dynamics, while ignoring low-level ion concentrations and such. We also simplify synapses into a simple voltage increase in the post-synaptic neuron with a hardwired or random delay and a tunable weight, see more in Section 2.4.2.

3.2.4 Anatomical accuracy of the model

A complete brain model would have to replicate the diverse regions in the brain, as introduced in Section 2.5. Some researchers have faithfully modelled multiple regions of the brain: Izhivevich in [23] accurately modelled the thalamocortical system, reproducing in detail the 6-layer structure of the cortical cirtuitry, the white matter region of cell bodies in the cerebrum as well as multiple thalamic nuclei. On top of this, the model included realistic connectivity and synaptic learning, making it a very complete model. The objective of his work was the creation of an accurate model, without the intent of obtaining particular dynamics. Our situation is different, as our main goal is to obtain human-like dynamics according to the neural correlates of consciousness, and we wish to construct the most simple model that can achieve these so as to understand the necessary components for conscious-like dynamics.

Instead of having different models for each anatomical region of the brain, we will make use of connectome data obtained from [18] to achieve a reasonable level of anatomical accuracy. More details on the connectome data will be given in Section 3.3.1.

3.2.5 Assumptions

To construct our simulation, we must specify which structures and functional units we aim to model and which we will forget about. In Section 2.5, we briefly discussed the anatomy of the human brain. Our purpose is to create models of the brain with dynamics that overall resemble those of conscious processes. Hence, we will not model all regions of the brain, but only the minimal set of structures to achieve our goal.

Firstly, we discard the peripheral nervous system; we assume that its input is somehow reflected in the dynamics of the brain, but do not care how exactly. This is not to say that the PNS is unrelated to consciousness. On the contrary, it provides sensory input to the brain and is crucial for the aspect of consciousness known as bodily awareness, which Anil Seth describes as one of the manifestations of consciousness in his TED talk: *Your brain hallucinates your conscious reality* [32]. Instead of modelling the PNS, we assume that some fraction of the input to our model comes from the PNS.

The cerebellum will also be ignored, as it is not typically associated with consciousness. The thalamus, however, is of great importance and is thought to be a center for the modulation of consciousness, as described in Section 2.5. The cortex, with its different areas, performs a range of sensory processing functions, and is therefore strongly related to the contents (e.g. visual) and level of consciousness.

3.3 Experiments

3.3.1 Connectome

As a starting point for a realistic network, we have turned to the literature in search of useful data and techniques. From advances in imaging and graph theory, we have seen the field of brain connectomics be born. The connectome is a map of neural circuitry in the brain, much like the genome is a map of the genetic sequence of an organism [4]. Hagmann et al. in their paper *Mapping the structural core of the human brain* used Diffusion Spectrum Imaging (DSI) to map neural pathways across the cortex and derive a map of structural connectivity in the brain [18]. Their study also analysed individuals' resting state brain activity through functional MRI (fMRI) and found a high correlation between structural and functional connections, which suggests to us that this structural map can be used as a basis for functionally realistic brain simulations.

The connectome resulting from the aforementioned study [18] consists of a network of 998 regions of interest (ROIs), a matrix of connectivity weights between the ROIs and the 3D location of each ROI within the brain. This structure helped us develop our next computational model of the brain: one that followed the connectome structure and placed a single neuron in every node of the network.

Figure 3.1 contains 4 different views of the 3D location of the 998 ROIs from the connectome.

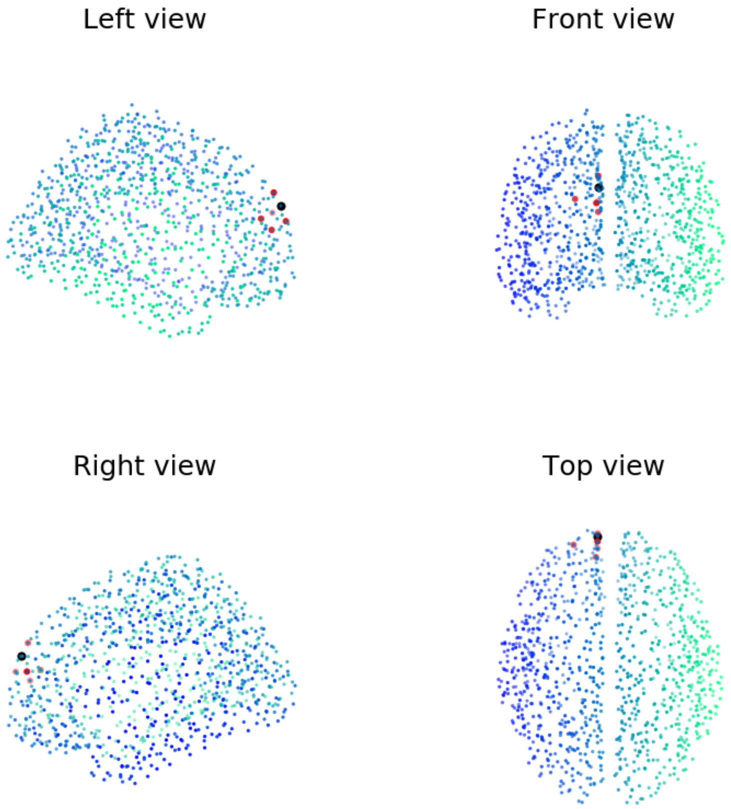


Figure 3.1: 3D view of the connectome, as obtained in the data from Hagmann et al. [18]. The color spectrum, which changes along a single axis, is used to help visualization. Each dot represents a ROI or node in the connectome, of which there are 998. The black dot is node 90, of relevance in Section 3.3.6. Red dots are strongly connected neighbours to this node.

Figure 3.2 shows the connectivity structure of the connectome data. The colorscale is associated to the base 10 logarithm of the weights and helps visualise the data.

3.3.2 Single-neuron modular network

To develop a working brain model from the connectome data [18], we initially decided to construct a network where every connectome node is represented as a single neuron, which would be either excitatory or inhibitory according to some probability distribution. To obtain a balanced network, it is crucial to get the balance between excitation and inhibition right. There is

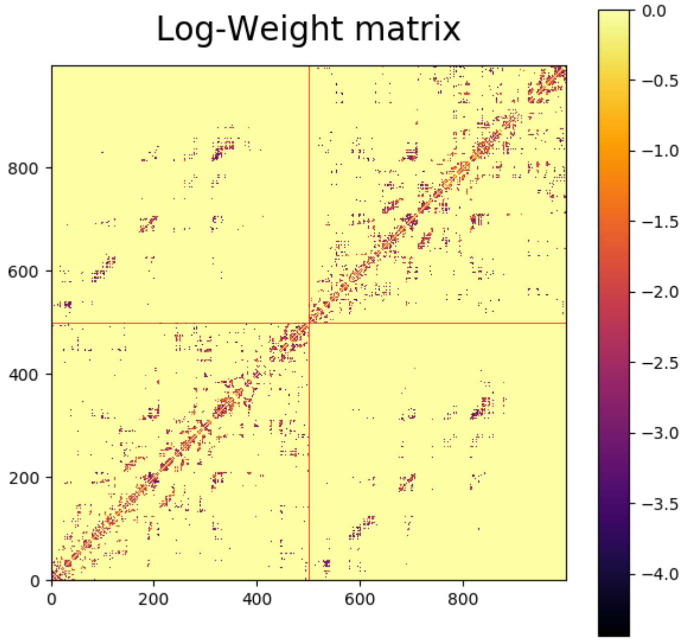


Figure 3.2: Logarithmic weight matrix of the connectome network

no perfect proportion between excitation and inhibition or a correct synaptic weight distribution, since this highly depends on the neuron models used, the number of neurons in the network and the structure of the same. Instead, we must experiment with our network until the combination of network structure, size and configuration of the neurons yields a model with realistic neural activity.

The main structural features of this network that we must configure are:

- Proportion of excitatory to inhibitory neurons.
- Connection density
- Synaptic weights.
- Synaptic delays.

We opted for a configuration with excitatory and inhibitory neurons on a 4 to 1 ratio, which means that ≈ 20% of the modules as inhibitory neurons and

≈ 80% as excitatory neurons, similar to what other researchers have chosen when modelling the mammalian cortex [22]. Excitatory and inhibitory neurons were assigned randomly to different nodes in the connectome. We then began to structure the network through synapses between the populations of neurons:

- **Excitatory to excitatory neuron synapses** were constructed according to the connectome matrix weights (with a tunable scaling factor) and with delays proportional to the euclidean distance between the nodes, by taking into account the transmission speed in the brain (approximately 2 m/s [30]).
- **Excitatory to inhibitory neuron synapses** consist of focal connections such that 4 excitatory neurons excite a single inhibitory neuron (in line with the 1 to 4 inhibitory to excitatory ratio). Synaptic weights between these two populations were sampled from a standard Gaussian distribution, and scaled up with a tunable scaling factor. Note that these synapses are excitatory synapses, since the spikes come from excitatory neurons; therefore, the post-synaptic neurons receive a positive increase in voltage after a pre-synaptic spike (See Listing 3.1).
- **Inhibitory neuron outgoing synapses** consist of diffuse connections, where each inhibitory neuron is connected to all other neurons. This system-wide inhibition is important in order to achieve oscillatory dynamics and synchronization between the populations of neurons. As before, the weights are sampled from a standard Gaussian but this time the synapses are inhibitory, and therefore the weights are negative (See Listing 3.1)

```
    excitatory_synapses = Synapses(
        excitatory_population, other_population,
        model = "w : volt",
        on_pre = "voltage += w"
    )
    inhibitory_synapses = Synapses(
        inhibitory_population, other_population,
        model = "w : volt",
        on_pre = "voltage -= w"
    )
```

Code Listing 3.1: Excitatory vs inhibitory synapses

As mentioned above, all the weights of the synaptic connections have tunable scaling factors. The values chosen for these are crucial for the dynamics of the network. As an example, if the excitatory-excitatory scaling factor (we

	EX_EX_FACTOR	EX_IN_FACTOR	IN_EX_FACTOR	IN_IN_FACTOR
Ranges of values	200-400	15-75	5-15	2-3
Configuration 1	400	15	8	2
Configuration 2	400	75	8	2

Table 3.1: Scaling factors for the synaptic weights between populations. The table shows the range of possible values and 2 configurations studied in Figure 3.5 and 3.6.

will refer to it as EX_EX_FACTOR from now on) is too small, the network will not sustain high activity, but if it is too high, the network will be fully synchronized, as the effect of just a few spikes will trigger all other neurons in the network to spike simultaneously, due to a cascade effect.

Searching for ranges of parameters that result in good dynamics was a challenging process. We iteratively narrowed down on ranges of values that did not result in too synchronized activity, but that instead showed signs of sustained, oscillatory and equilibrated activity. Figure 3.3 shows how the mean firing rate of the excitatory population varies as we alter the scaling factors and hence the magnitude of the synaptic weights between the neurons. An example of a fully-activated network, whose dynamics are not interesting due to its chaotic nature can be seen in Figure 3.4, where a low IN_IN_FACTOR leads to insufficient inhibition in the network.

From the tuning step in Figure 3.3, we came up with a range of values that we can use for our network. These are shown in Table 3.1. Now we can take a look at some of the resulting dynamics and power-frequency distributions of the network activity for different configurations.

In Figure 3.5, we observe some sort of coalition formation taking place, marked by synchronized firing of groups of excitatory neurons, and oscillatory activity that can be clearly distinguished by the repeated patterns in the raster plot, or by the peaks and troughs in the firing rate plot. We can see that there is a peak in the power-frequency plot at 5Hz, then smaller peaks around 9Hz, 12Hz, 14Hz, 17Hz and some presence of higher-frequency oscillations around 28Hz and 32Hz. These are interesting dynamics, where Theta (4-8 Hz) and Alpha (8-15 Hz) band oscillations are dominating, and there is some sign of Gamma (30-80 Hz) band oscillations.

Now let's take a look at another configuration where we have increased the inhibition in the network by supplying a larger value for the IN_EX_FACTOR. This effectively results in the excitatory activity being suppresed faster, since the connections towards inhibitory cells are now stronger. Figure 3.6 shows

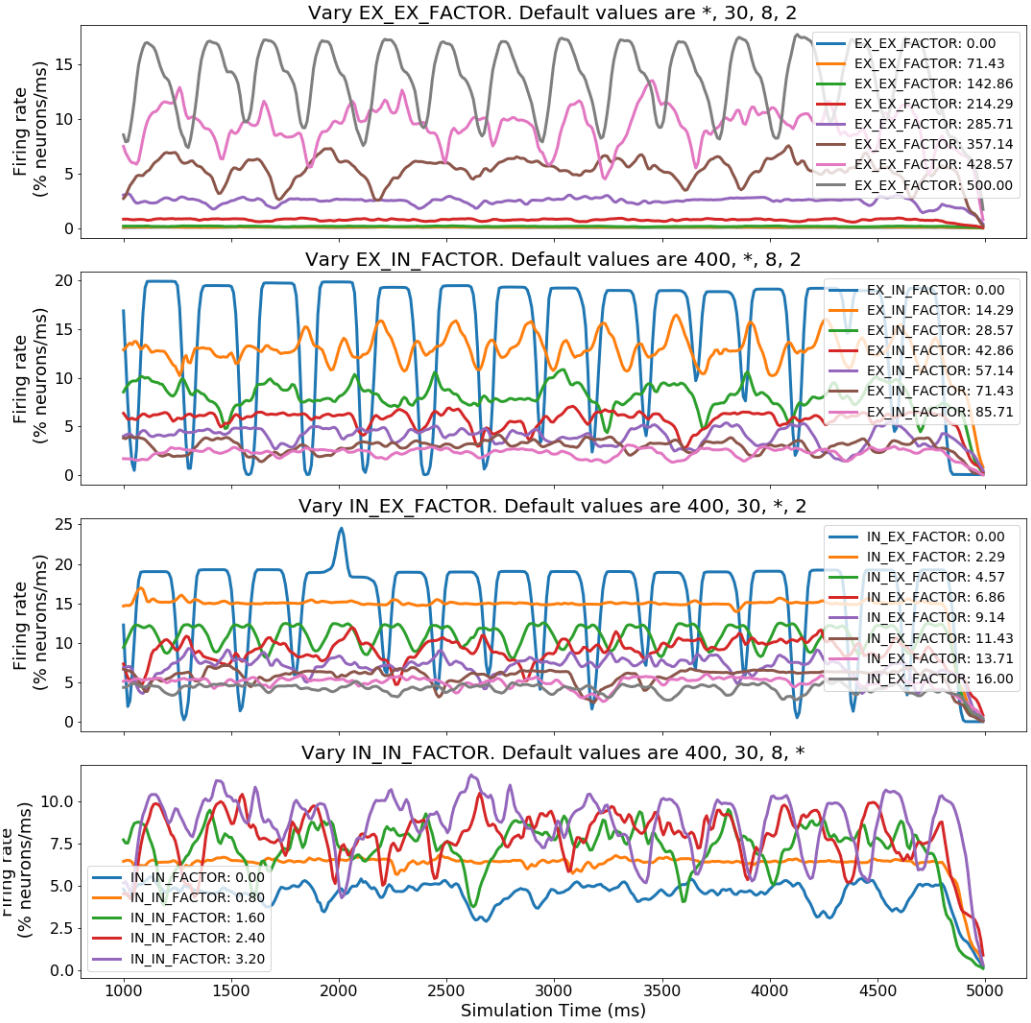


Figure 3.3: Tuning of the scaling factors in the single neuron connectome network. These firing rates correspond to the excitatory neurons in the network and are calculated as a moving average of the instantaneous firing rate with an averaging window of 200ms and measurements taken every 10ms. On each averaging window, the total number of spikes is counted and divided by the averaging window length, giving the instantaneous firing rate per millisecond, and finally the values are divided by the number of excitatory neurons, resulting in the percentage instantaneous firing rate. When a value is tuned, the other scaling factors are set to a reasonable default.

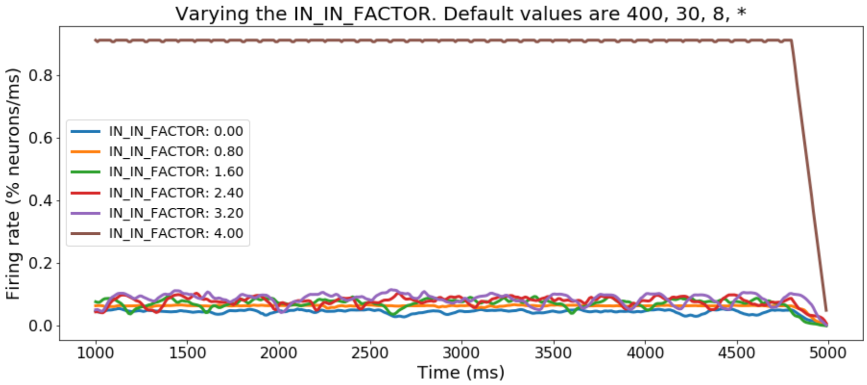


Figure 3.4: Parameter tuning with high `IN_IN_FACTOR` leading to full-network activation and uninteresting dynamics for `IN_IN_FACTOR` = 4.0.

the activity for this network setup. Figure 3.6 presents 3 seconds of activity, where we observe that the inhibitory population is always at the same activation level as the excitatory one, contrary to what we saw in Figure 3.5, where the inhibitory firing rate was always low, although the peaks coincided with peaks in excitatory activity. In this configuration, we see much lower activation of the network (compare 30% peaks in the firing rate in Figure 3.5 vs 7% firing rate peaks in Figure 3.6), as a result of the increased inhibition, and we also see what seems like more irregular activity. The power-frequency diagram shows an exponential decrease in power as the frequency increases (linear decrease in log-linear plot), with all the power concentrated in the Theta (4-8 Hz), Alpha (8-15 Hz) and lower Beta (15-30 Hz) bands and almost no power in the higher frequencies.

This experiment helps to demonstrate the wide range of dynamics that we can generate by modulating the activity of the network in different ways. Another aspect that we might want to tune is the synaptic delay between the different populations. In the above configurations, we opted to use the euclidean distances between nodes and a constant transmission delay throughout the brain of 2 m/s to derive synaptic delays for the excitatory-to-excitatory synapses. For the rest of focal and diffuse connections we chose constant delays of 1ms, mainly because the inhibitory neurons would otherwise not be able to synchronise distant regions in the brain. In later experiments, this will not be an issue, because the network will contain modules of neurons which are geographically close. Therefore, short and constant delays will be appropriate.

Although simple and convenient, it is unrealistic to model the brain as a

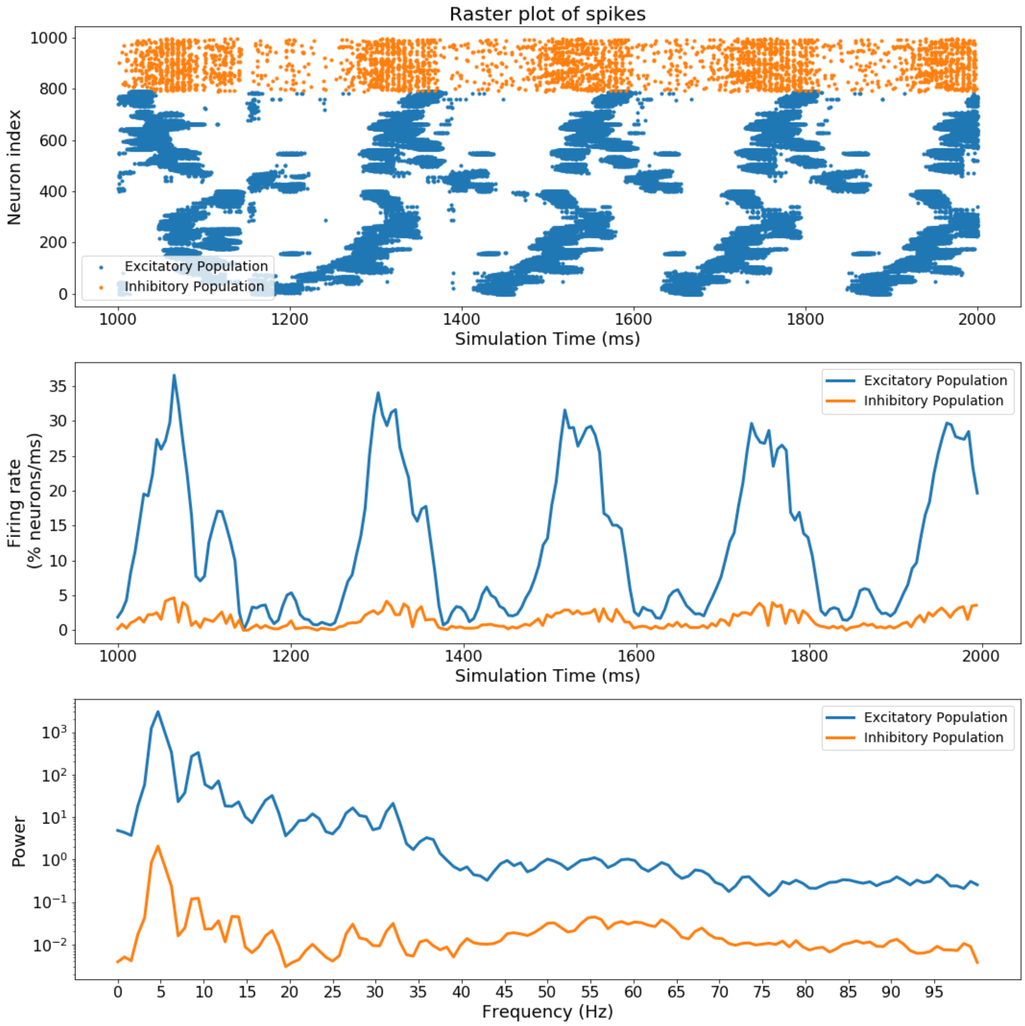


Figure 3.5: Raster plot, firing rate plot and power-frequency plot of the single-neuron modular network simulation, run for 20 seconds (1 second shown). For this simulation, we set the scaling factors to 400, 15, 8 and 2, as shown in Configuration 1 in Table 3.1. The firing rate plot was constructed with an averaging window of 5ms and measurements every 5ms.

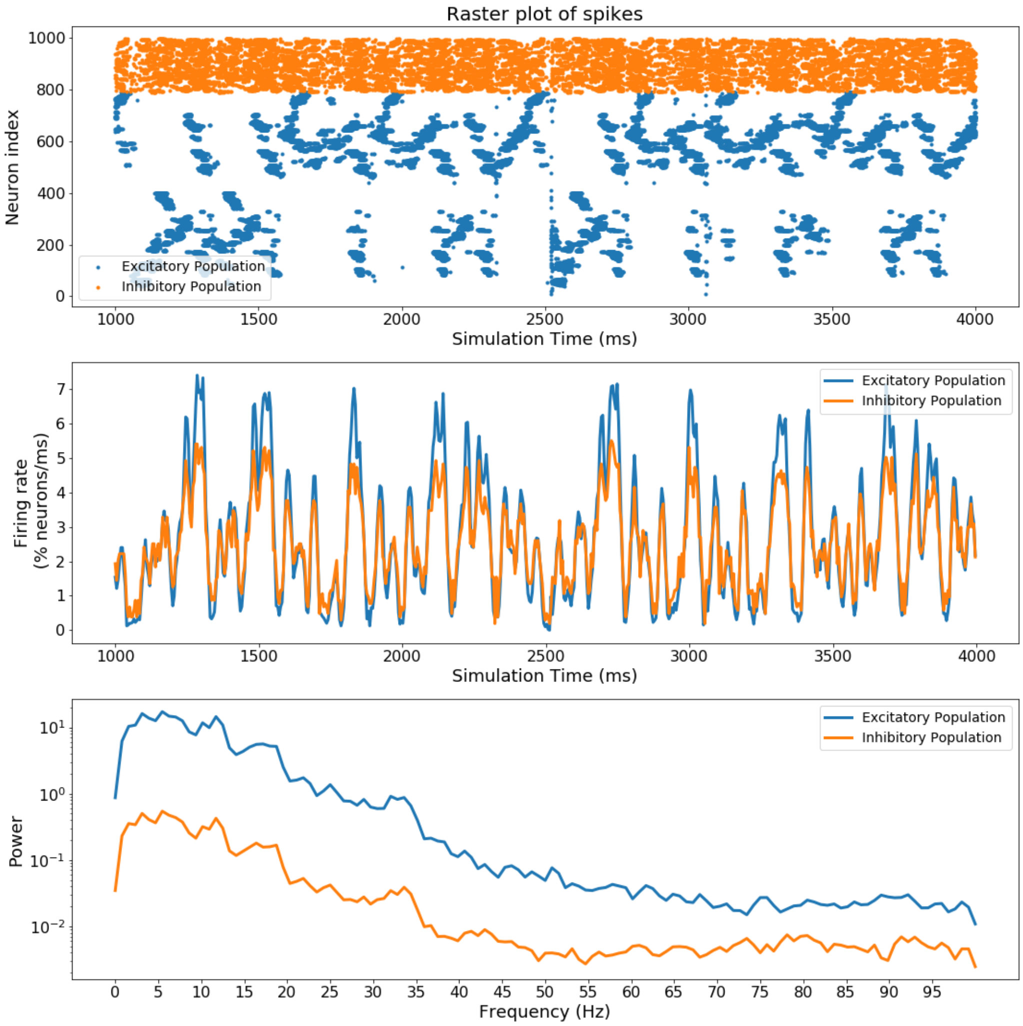


Figure 3.6: Raster plot, firing rate and power-frequency plot of the single-neuron modular network simulation, run for 20 seconds (3 seconds shown). For this simulation, we set the scaling factors to 400, 75, 8 and 2, as shown in Configuration 2 in Table 3.1. The firing rate plot was constructed with an averaging window of 5ms and measurements every 5ms.

mere network of connections without any further inner structure. The brain is known for exhibiting widely distributed and distinct activity throughout its multiple regions, while at the same time this activity is coordinated and integrated. These features require a higher level structure within the connectome map, which in our case is achieved by modelling each module or node of the connectome as a small network of neurons. Each module will be a small balanced network, the modules will all be connected by inter-modular excitation according to the connectome map. However, before we proceed to build this type of network, we need to come up with an internal modular structure that helps to reflect brain dynamics. As such, we will now focus on the design of a small individual module in this network.

3.3.3 PING oscillatory module

Synchronization across oscillating networks in the brain is thought to be the key to understanding the transition from single-neuron dynamics to high-level behaviour [7]. We could create a small network of neurons and interconnect them randomly, such that it sustains elevated activity without becoming saturated. However, the dynamics of this network would not follow patterns of activity that have actually been oberved in the brain. Instead, let's create modules with realistic, oscillatory dynamics like we might find in the cerebral cortex. A realistic setup, that leads to oscillatory gamma-band dynamics can be achieved through a structure known as PING, which stands for Pyramidal InterNeuron Gamma. This setup envisions a network with two populations, one of excitatory neurons and another with inhibitory neurons, interconnected in the following way:

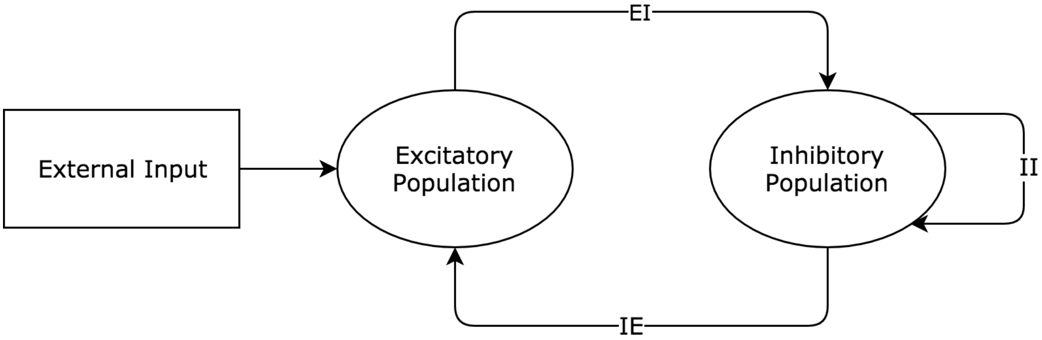


Figure 3.7: PING module architecture. [5]

With the PING architecture, we obtain networks which oscillate in the

Gamma-band (30-80 Hz), and we can tune the frequency of oscillation by modifying the conduction delay between populations. Bhowmik and Shanahan demonstrated in their paper *Metastability and Inter-Band Frequency Modulation* how this type of network architecture can be set up to achieve a wide range of oscillatory behaviours. Inspired by their work, we have constructed a network of 50 neurons (40 excitatory and 10 inhibitory) that display Gamma-band oscillations at 50Hz. Figure 3.8 presents a summary of the dynamics of our network.

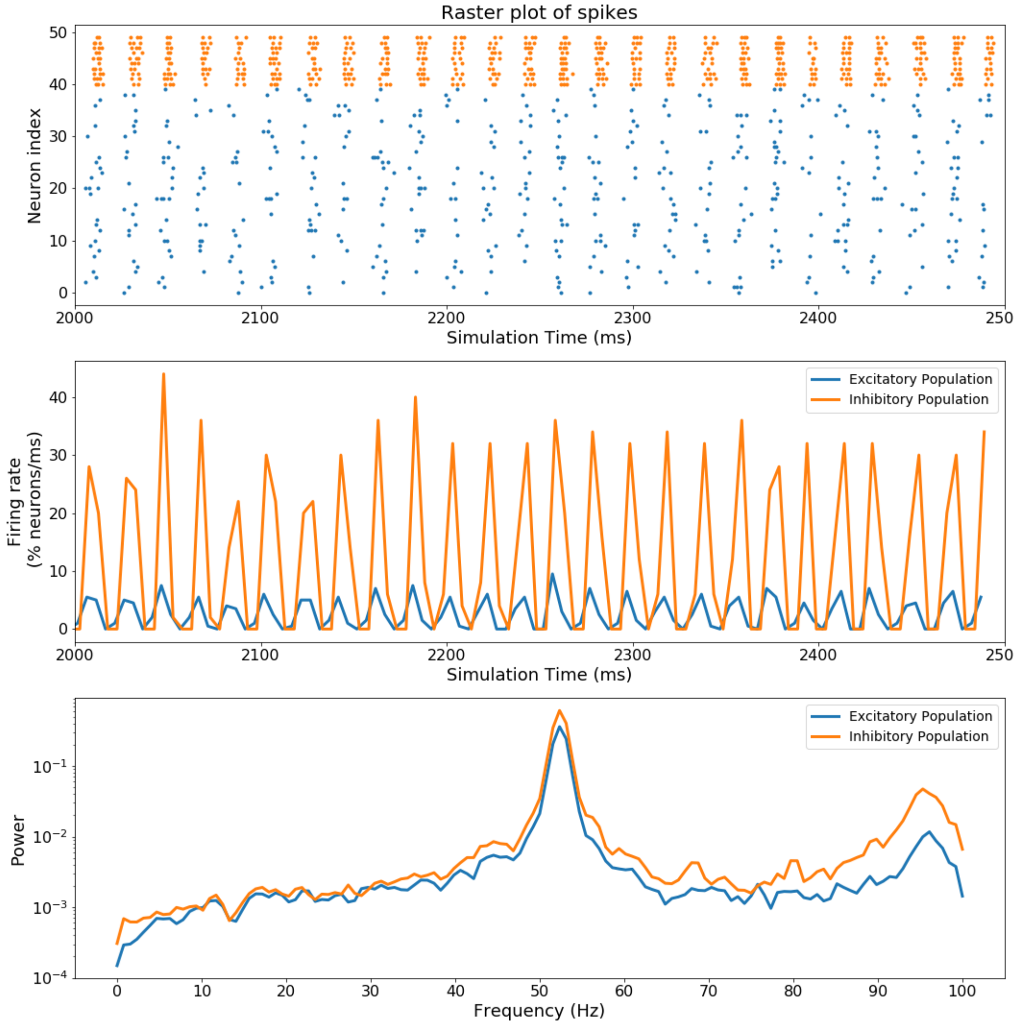


Figure 3.8: Network dynamics of a PING module with 50 neurons. The network oscillates at 50Hz, as shown by the peek of the power-frequency plot.

Figure 3.8 reveals the oscillatory behaviour we are discussing. In the top raster plot, excitatory neurons fire in a syncronized fashion, followed by a period of no activity. This is a result of the inhibitory neurons quickly reacting to the excitatory activity and avoiding any more excitatory firing for a short period of time. The middle firing rate plot helps us see the level of regularity in this oscillation in terms of the number of neurons spiking, forgetting about the neuron indexes. Finally, the lower plot is obtained through `Scipy`'s `signal.welch` method, which performs the Fourier transform on the firing rate time series to obtain an approximation of the power-frequency distribution of the spiking data (See `https://docs.scipy.org/doc/scipy-0.14.0/reference/generated/scipy.signal.welch.html`). This power-frequency plot has a prominent spike at 50Hz, the fundamental frequency of oscillation, and a smaller spike around 100Hz, probably corresponding to the first harmonic.

Network architecture

Our network consists of 50 neurons: 40 excitatory and 10 inhibitory neurons, instead of the 250 neurons used in [5]. In later experiments, we will construct a network according to the connectome architecture (1000 nodes) and a PING oscillatory module in each node, resulting in a total of $50,000$ neurons. The computational load to model $1000 \times 250 = 250,000$ neurons instead is too large (as we will later see), hence the reason for reducing the size of the PING modules.

The network is driven by external stochastic input to the excitatory neurons following a Poisson distribution with rate $\lambda = 20Hz$, and is scaled up by a factor of 8, as in [5]. The Possion rate is larger than the one used in [5] (recall that the poisson rate is the mean number of events ocurring within a time interval), to account for the smaller number of neurons: we need more input to our network to have the same degree of activation because there are less synapses between the neurons, leading to less activity.

Although it may seem straightforward to achieve these oscillatory dynamics, it requires a great deal of fine-tuning the synaptic weights and delays between the neuron populations. Table 3.2 summarizes the parameters used for the network. These parameters are in careful balance and small variations will alter the frequency as well as the regularity of oscillation. For instance, increasing the delays leads to lower frequencies of oscillation and viceversa. On the other hand, lower connection probabilities lead to less synchronized oscillation, which can be seen as more irregularity and the oscillatory behaviour becomes less clear to the human eye.

	Connection probability	**Weights**		**Delays**	
		Mean	*SD*	*Mean*	*SD*
EI	70%	10	2	2	1
IE	100%	10	2	5	2
II	100%	10	2	5	2

Table 3.2: Configuration of PING module for 50Hz oscillation, including means and standard deviations (SD) for the normal distributions from which synaptic weights and delays between neuron populations are sampled.

3.3.4 Modular connectome oscillatory network

Now we have all the components for a full-scale modular network. The PING oscillatory modules developed in Section 3.3.3 will be the building blocks for our complete model, which will integrate these modules with the connectome described in Section 3.3.1 to shape the connections between these modules.

The resulting architecture consits of 998 modules of PING oscillation, each containing 40 excitatory neurons and 10 inhibitory neurons, for a total of $49,900$ neurons. The network receives poisson distributed stochastic input to the excitatory neurons with a frequency of 20 Hz and is scaled up to 8 mV, as in Section 3.3.3, ensuring sufficient network wide activation.

In order to provide a wide range of dynamics throughout the network, we have introduced variability across the modules. It would not be realistic to have all modules oscillate at the same frequency and with the same regularity; instead, we will design the modules with different frequencies of oscillation. To do this, we let each module have an internal random mean delay sampled from a uniform distribution between 1 ms to 15 ms and weights sampled uniformly between 5 and 15 mV/ms. The range of delays allows the frequency of oscillation of the module to vary from 30-35 Hz (with a large delay) all the way to 80-90Hz for the shortest delays. The weights, on the other hand, modulate the degree of activation of the neurons in the module and are reflected in the regularity of firing of the neurons: big weights lead to regular behaviour and very active modules, whereas the smaller weights correspond

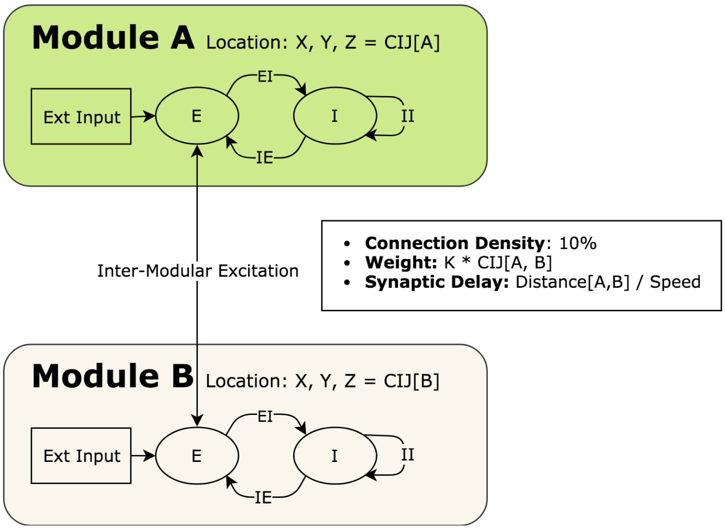

Module A Location: X, Y, Z = CIJ[A]
Ext Input
E
EI
I
II
IE
Inter-Modular Excitation
Connection Density: 10%
Weight: K * CIJ[A, B]
Synaptic Delay: Distance[A,B] / Speed
Module B Location: X, Y, Z = CIJ[B]
Ext Input
E
EI
I
II
IE

Figure 3.9: Structure and interconnections between modules. Ovals represent populations of neurons, either excitatory (E) or inhibitory (I). Arrows represent synapses from one neuron population to another.

to irregular oscillators. The delays and weights we refer to are the synaptic delays and weights interconnecting the excitatory and inhibitory populations of neurons within each module, which in Figure 3.9 are the EI, IE and II synapse groups.

Here is an example of a simulation being run with the following command:

```
import modular_connectome_experiment as ex
data = ex.run_experiment(
    n_mod=1000,
    duration=20000,
    inter_scaling=50,
)
```

Here is the output of a sample simulation, run on a single core of a laptop with a 2.9 GHz Intel Core i7 processor with 16GB of LPDDR3 RAM:

```
Setting up synapses...
    EX_IN_SYN... (279,363 synapses) [0.97s]
    IN_EX_SYN... (399,200 synapses) [1.18s]
    IN_IN_SYN... (99,800 synapses) [0.34s]
    INTER_EX_EX_SYN... (5,713,203 synapses) [32.74s]
All synapses created [35.24s]
```

```
Supplying Poisson input to network... [0.00s]
Running sym... [881.00s]
```

As we can see, the simulation consists of 1000 modules (in fact it's 998), lasts for 20 seconds (20,000 milliseconds) and we have chosen the inter modular scaling factor to be 50, a value that causes modules to affect each other's activity without completely syncronising their activity. Let's look at some statistics about this experiment:

```
Simulation time: 20 seconds.

Number of spikes:
    Excitatory: 31,496,686 spikes
    Inhibitory: 22,694,911 spikes
    Total: 54,191,597 spikes

Mean firing rate of the network:
    Excitatory population: 3.94%
    Inhibitory population: 11.37%
    All: 5.43%
```

These numbers help us get a sense of the scale of the simulation, but in fact tell us nothing about how well this network models the brain. To do so, let's move on to study the dynamics of this network.

Dynamics

The activity of this network is widely differentiated, thanks to the variability we have introduced, meaning that there is a variation in the frequencies seen throughout the brain. The frequency spectrum of the activity follows a straight line with negative slope in a semilog plot (see Figure 3.10), as is typical of sleep brain EEGs, although the lower frequencies are missing.

Since we have already studied the individual PING modules that make up this experiment in the single-neuron modular network in Section 3.3.2, what is now interesting is to observe how the interconnections between these modules affect the network dynamics.

The weights between modules are created according to the connectome, which is described in Section 3.3.1. These weights, however, are scaled by a factor which is tunable in our model. This scaling factor is applied in the simulation as shown in Listing 3.2 with the variable `inter_scaling`.

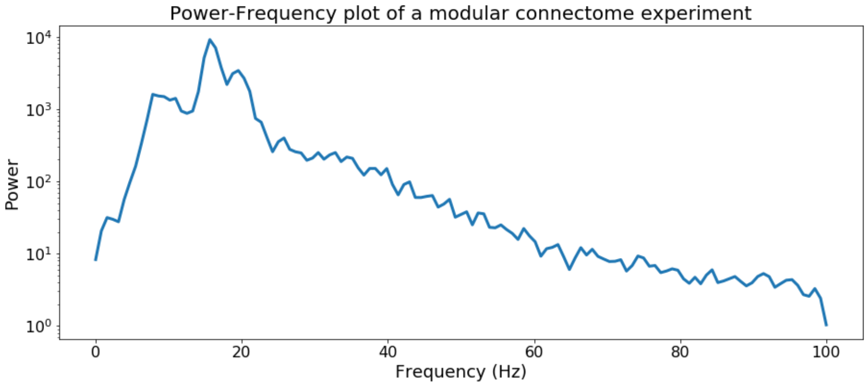


Figure 3.10: Power-frequency spectrum for a 20 second simulation of the complete modular connectome experiment. Inter-modular synaptic weight scaling factor of 50.

```
# Synapse group for synapses between excitatory
# cells across modules.
INTER_EX_EX_SYN = Synapses(EX_G,
    model='w: volt',
    on_pre='v += w',
)
synapses, delay_matrix = get_connectivity(
    n_mod, n_ex_mod, inter_conn, XYZ, CIJ
)
# Split synapses into source and target neuron index
synapses_i, synapses_j = \
    [np.array(s) for s in zip(*synapses)]

# Neuron indices to join
INTER_EX_EX_SYN.connect(
    i=synapses_i,
    j=synapses_j
)

# Synaptic delays between each pair of neurons.
INTER_EX_EX_SYN.delay = delay_matrix[
    # Get module number for each neuron index
    synapses_i / n_ex_mod,
    synapses_j / n_ex_mod
] * ms

# Synaptic weights between each pair of neurons.
INTER_EX_EX_SYN.w = CIJ[
    synapses_i / n_ex_mod,
    synapses_j / n_ex_mod
] * inter_scaling * mV
```

Code Listing 3.2: Inter-modular synapse setup

When the inter-modular scaling factor is small, the modules behave largely independently and the mean network activation is small. However, as this interconnection factor is increased, the modules begin to influence each other's activity significantly and eventually the whole network oscillates in a synchronous fashion. This effect is clearly demonstrated in Figure 3.11, where we vary this factor from 0 to 100. Not only do the modules begin to activate syncronously when the scaling is big, we also see a network wide activation when it happens, instead of the low mean network activation associated with smaller scaling values.

Results

Figure 3.11 demonstrates that our model correctly reproduces the behaviour of cortical synchronization described in 2.2.5, where a reduction of cortico-cortical synaptic strength is associated with the loss of network wide synchronization and with the transition from sleep to wakefulness. In addition, we also observe the progression from high-magnitude low-frequency delta waves to low-magnitude high-frequency oscillations observed in human EEGs during the sleep to arousal transition [35]. The low-frequency oscillations are slightly faster than they typically are during deep sleep, and are certainly not SWA (< 1Hz). This may be evidence that slow, long-lasting processes cannot be correctly modelled by this type of simulation (See discussion in the Conclusion Section 4).

3.3.5 Thalamus-driven network

Our complete thalamocortical model requires the introduction of a thalamus-like structure to it. As described in Section 2.5, the thalamus is a structure that relays information to the rest of the cortex and is therefore connected to all areas of it. The thalamus is also closely related to sleep homeostasis and is therefore crucial in regulating the level of consciousness in the brain.

Before going into the details of the experiment, we provide a necessary context on the methodology for calculating Lempel-Ziv complexity in a simulation.

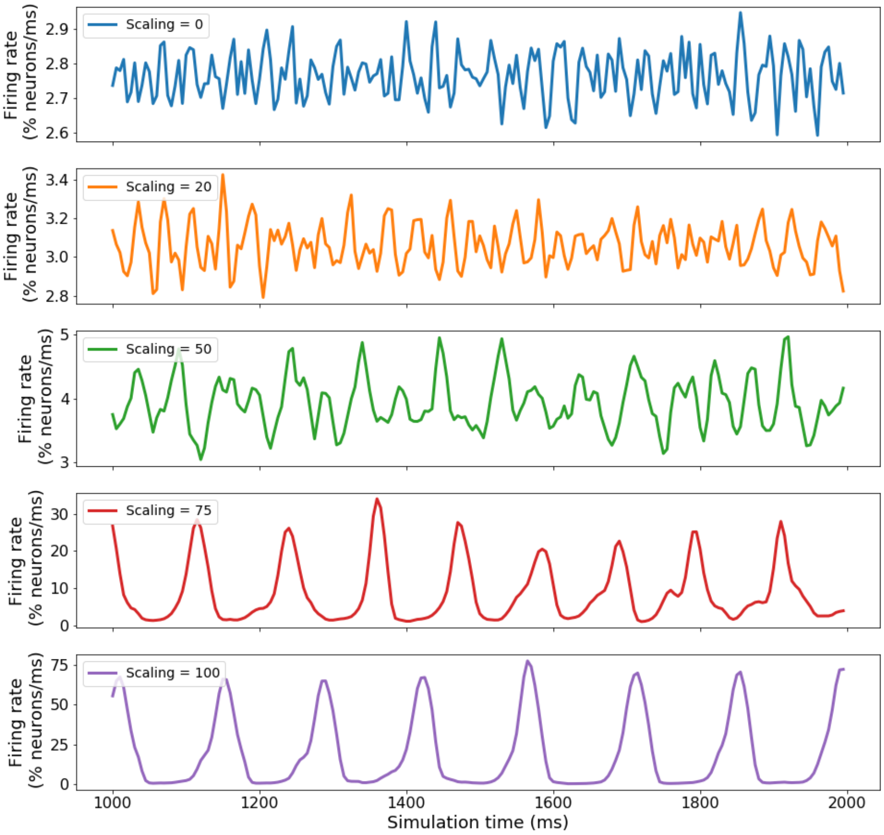


Figure 3.11: Mean firing rate of the network for different scaling factors between modules

Calculating Lempel-Ziv complexity

We will analyse the LZ complexity of the dynamics of each module in the network to obtain a distribution of values summarising the complexity of the whole simulation.

We take the following steps to obtain the complexity distribution of a simulation:

1. Calculate a moving average of the instantaneous firing rate of each module. The moving average firing rate algorithm is straightforward,

and can be found in the code archive associated with this submission.

2. Binarize the moving average time series by converting above average rates to ones and the rest to zeros. This is similar to how PCI computes the matrix of significant sources from the EEG data.

3. Apply the LZ76 algorithm to obtain the complexity of each of the binarized time series, obtaining a total of 998 measures, the distribution of which we can plot to get an idea of the simulation's complexity. We used the LZ76 implementation from [24], which can be found in the code archive.

This schema is inspired by the PCI measure, where a spatiotemporal matrix of significant activation sources is constructed before applying LZ76 on it, as discussed in Section 2.2.4.

Experiment

The starting point for this experiment is the same connectome network used in the previous section (Section 3.3.4), which consists of a modular connectome network and variabile dynamics across the modules. The only modification is the introduction of a special module of 250 neurons (200 excitatory and 50 inhibitory) that models the thalamus and oscillates at some random Gamma-frequency. Note that this frequency of oscillation is perhaps not realistic for the thalamus and in future designs of the model we shall investigate in more depth how to accurately model the thalamus.

We have chosen to model the thalamus as an oscillator that is outwardly connected to all modules in the network and whose effect on the network can be modulated. This model of the thalamus is of course very simplified, and in fact lacks one important feature, namely that in the real brain there are both corticothalamic (outward) and thalamocortical connections (inward). Our purpose here is not to accurately mimic the human thalamus, but rather to include a structure in our model that has this global connectivity structure onto the cortex, which is similar in some ways to the thalamus.

We ask ourselves: what changes in activity do we expect to see after including the thalamus as part of our modular network? If a central structure is simultaneously stimulating different areas and regions in the brain, those areas will behave more similarly, since they receive stimuli with the same timings, leading to synchronised activity and less differentiation, as occurs during sleep.

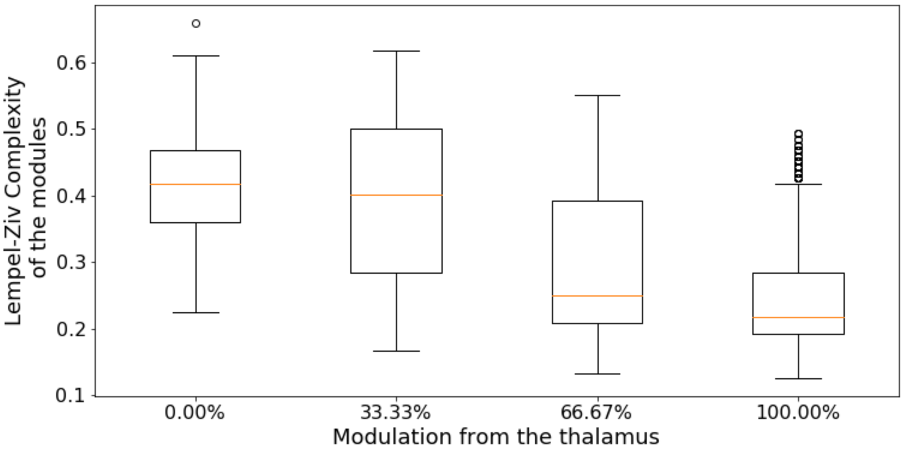


Figure 3.12: LZ complexity of the modules from simulations with varying modulation from the thalamus. The scaling factor for inter-modular synapses was set to 20.

This result should be strengthened with an analysis of the complexity of the neural activity. We hypothesised that the LZ complexity of the network activity would decrease as we increase the modulation of the thalamus effect on the rest of the network.

Results

Figure 3.12 shows a generalized decrease in LZ complexity in the modules as the modulation from the thalamus increases, especially comparing the unmodulated simulation with the 66.67% and 100% modulation experiments. However, it is also important to measure the statistical significance of this difference in dynamic complexity. With the help of a statistical test, we can compare the complexity of a random module from one simulation with that of another.

This is precisely what the Mann-Whitney U Test allows us to measure. Let a sample be the list of LZ complexity values from a simulation, as explained in the above Section 3.3.5. The Mann-Whitney U Test is a non-parametric test of the null hypothesis that a random module from a sample is equally likely to have less or greater complexity than a random one from another sample, i.e. that the samples have the same distribution. As the alternative hypothesis, we would propose that a value from the first group has higher

	0% vs 33%	33% vs 66%	66% vs 100%	0% vs 100%
p-value	0.0017	10^{-78}	10^{-25}	10^{-227}
U statistic	535,576	738,450	631,247	911,758.5

Table 3.3: p-values and U test statistics from the Mann-Whitney U test when comparing the LZ complexity of the modules between different simulations. The compared simulations vary the degree of modulation from the thalamus, from 0% to 100%. Each simulation results in a distribution of LZ complexity values, as shown in Figure 3.12.

complexity than one from the second group. In our context, rejecting the null would translate to saying that a weakly modulated simulation is associated to higher complexity values than strongly modulated one.

This test does not require the assumption that our data is normally distributed, which is helpful in our case. The Mann-Whitney U Test is available in `Python`'s `scipy.stats` package, returning the test statistic U, which counts, out of all NxN possible pairings of values, in how many of them the first value wins (is greater) over the second; and the p-value giving the probability of observing data at least as extreme as our data [39].

At a 5% significance level, the p-values in all 5 tests are lower than our significance level $\alpha = 0.05$ (see Table 3.3 for the p-values), therefore we reject all the null hypotheses that the distributions between the different simulations have the same distribution of LZ complexity (pairwise), in favor of the alternate hypothesis that the complexity of the first sample is greater than that of the second for all 4 tests.

As we initially hypothesised, we have concluded that indeed the dynamical complexity of the network decreases as we increase the modulation from the thalamus module, similarly to how it occurs during loss of consciousness.

3.3.6 Perturbational Complexity Index experiment

We are interested in reproducing the effects of the Perturbational Complexity Index (PCI) on a computational model of the brain. As explained in the background section (Section 2.2.4), PCI was part of a clinical study where the brains of patients were stimulated using transcranial magnetic stimulation (TMS) and the spatiotemporal pattern of activation in the brain was recorded, converted into a binary matrix of significant activation sources and compressed using LZ complexity to finally obtain the PCI NCC.

Experiment

The first step of this experiment is to simulate TMS on our computational model. According to Esser et al., a TMS pulse typically affects the cortex in a region of 10 to 20 mm in diameter [14]. The average size of the ROIs in the connectome is of 1.5cm^2 [18], which would mean that a TMS pulse would not stimulate all the neurons in a single ROI. In fact, comparing the approximate volume of an ROI $\approx \mathrm{O}(1\ \text{cm}^3)$ with that of the TMS effect region $\approx \mathrm{O}(10^{-3}\text{cm}^3)$, we see that only a thousandth of all neurons in the ROI would approximately be affected by the pulse. However, since our brain model is not real-scale and our modules are made up of only 50 neurons, stimulating a thousandth of the module would mean stimulating less than a single neuron, which would have no effect on the network. Instead, we will model the TMS pulse as a syncronous activation of a whole ROI. In addition, we will choose a ROI in the convex hull of the 3D connectome map, since TMS affects only brain areas close to the scalp.

To simulate a TMS pulse, we added a `SpikeGeneratorGroup` to the model that simultaneously supplied spikes to all excitatory neurons in a module for a sustained period of time, which we chose to be 50 milliseconds (shorter duration has too little of an effect).

Listing 3.3 demonstrates the creation of the neuron group simulating TMS, which will then be connected to a specific module of the model through the creation of one-to-one excitatory synapses from this `TMS_G` neuron group.

```python
TMS_G = SpikeGeneratorGroup(
    # Number of neurons affected by TMS
    n_tms,

    # Neuron indexes that spike [0, ..., N, 0, ..., N, ...]
    # Repeated as many ms as the tms stimulus lasts
    np.concatenate([
        np.arange(n_tms)
        for _ in range(tms_duration)
    ]),

    # Spike times associated to the neuron indexes above.
    # We generate spikes at:
    # [tms_t, tms_t + 1, ..., tms_t + tms_duration - 1]
    np.concatenate([
        [tms_t / ms + k for _ in range(n_tms)]
        for k in range(tms_duration)
    ]) * ms
)
```

Code Listing 3.3: `SpikeGeneratorGroup` simulating the effects of TMS

The module where TMS is pointed to was module 90 from Hagmann et al.'s connectome, which lies in the convex hull of the connectome, where TMS would realistically be able to produce an effect.

Given this computational setup, what remains is to implement the PCI complexity measure and to test the effects of TMS on PCI as our model transitions from wakefulness to sleep, as modelled in the modular connectome experiment in Section 3.3.4 (See Figure 3.11).

Calculating PCI

In the original PCI paper, the authors perform source modelling and non-parametric statistics to convert the EEG data into a matrix of spatiotemporal significant activation sources. This approach was inspired by their own previous research, where they presented a way to characterise the electrical responses of TMS on the cerebral cortex [8]. To do so, they analysed 300ms of post-stimulus firing rates and compared it with pre-stimulus baseline activity and determined both spatially and temporally whether each source presented statistically-significant activation.

Guided by this approach, but in a simplified way, we compared baseline activity of the brain with the 300ms of post-stimulus activity. After computing the pre-stimulus base firing rate of each module, the post-stimulus firing rates of the modules where binarized as 1s when the firing rate was two standard deviations above from the pre-stimulus baseline average and as 0s otherwise. Listing 3.4 shows a summarized snippet of the code where this PCI approximation is computed.

```
def pci(...):
    # t_pre, t_post: time before and after TMS to analyse.
    # dt, shift: averaging window and time between
    #       measurements in moving average computation.

    t1, t2, t3 = t_tms - t_pre, t_tms, t_tms + t_post

    # Number of time steps of post-TMS data
    post_ts = t_post / shift

    # Spatiotemporal matrix of activation
    post_data = np.zeros(n_mod * time_steps)

    for mod, mod_spikes in spikes_grouped_by_modules:

        # Pre-TMS moving-average firing rate
        pre_ma, _ = psd.moving_average(
            mod_spikes, dt, shift, t1, t2
        )

        # Activation of a module when activity is above
        # the threshold.
        threshold = pre_ma.mean() + 2 * pre_ma.std()

        post_ma = psd.moving_average(
            mod_spikes, dt, shift, t2, t3
        )
        post_data[mod * post_ts :(mod + 1) * post_ts] = \
            (post_ma > threshold).astype(int)

    # PCI = Normalized LZ complexity of spatiotemporal
    #       matrix of activation
    norm = np.log(len(post_data)) / len(post_data)
    return LZ76(post_data) * norm
```

Code Listing 3.4: Implementation of an approximated PCI measure.

Results

In Figure 3.13, we compare TMS in wakefulness (Scaling = 50) and LoC (Scaling = 100) models. Visual inspection of the graphs indicate that there is no clear effect caused by TMS. We see some variation in the dynamics of the upper two graphs, but no evident change in terms of baseline activity, and absolutely no effect in the bottom two simulations.

We then compared the firing rates of the modules where the TMS effect is the highest (the module that receives TMS and the modules more strongly connected to it according to the connectome weights) with the rest of the network. Module 90, where TMS is targeted, is most strongly connected to modules 95, 86, 87, 83, 91. The location of these modules can be seen in the connectome Figure 3.1, marked in larger black and red dots. Figure 3.14 compares the TMS effect on these modules with the activity of the whole network. The neighbours' activity slightly synchronizes with module 90, but

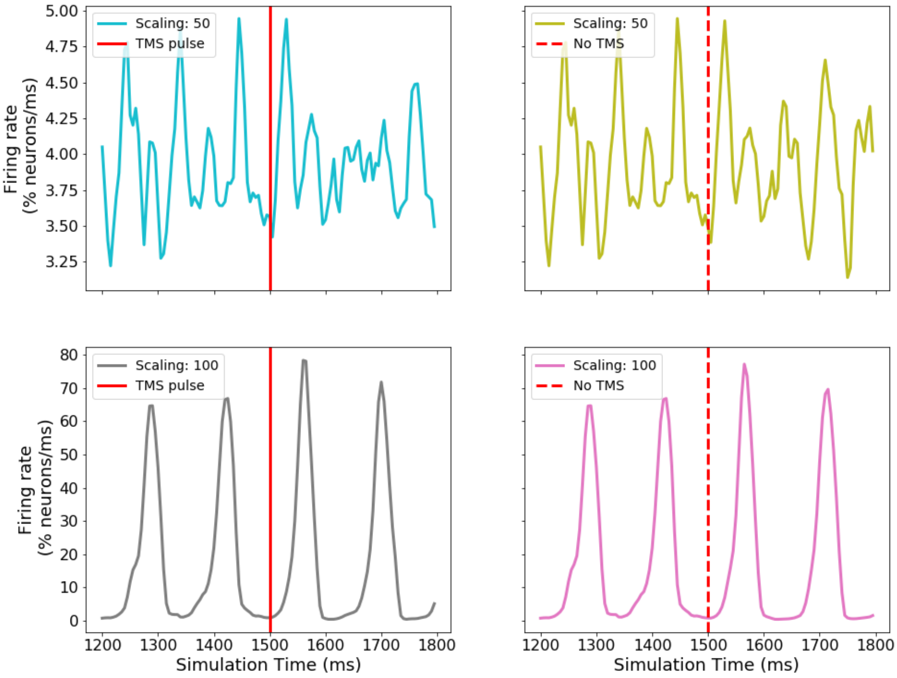


Figure 3.13: TMS stimulus to wakefulness and LoC computational models.

the overall activity remains unaffected, as seen in the bottom plot of this figure.

The data indicates that TMS indeed stimulates synchronously the affected module, as well as those strongly connected to it, but overall, it has a neglibile effect on the network. Most importantly, the effect fades almost as soon as the TMS pulse disappears, thus resulting in a non-lasting overall effect. We will now proceed to compare the actual results obtained through PCI, but we already know that temporally speaking, the complexity of the dynamics caused by TMS are insatisfactory.

	TMS	No TMS
Scaling = 50	0.182	0.175
Scaling = 100	0.125	0.118

Table 3.4: PCI values for wakefulness and LoC simulations with and without the presence of a TMS pulse. Values respective to the plots in Figure 3.13.

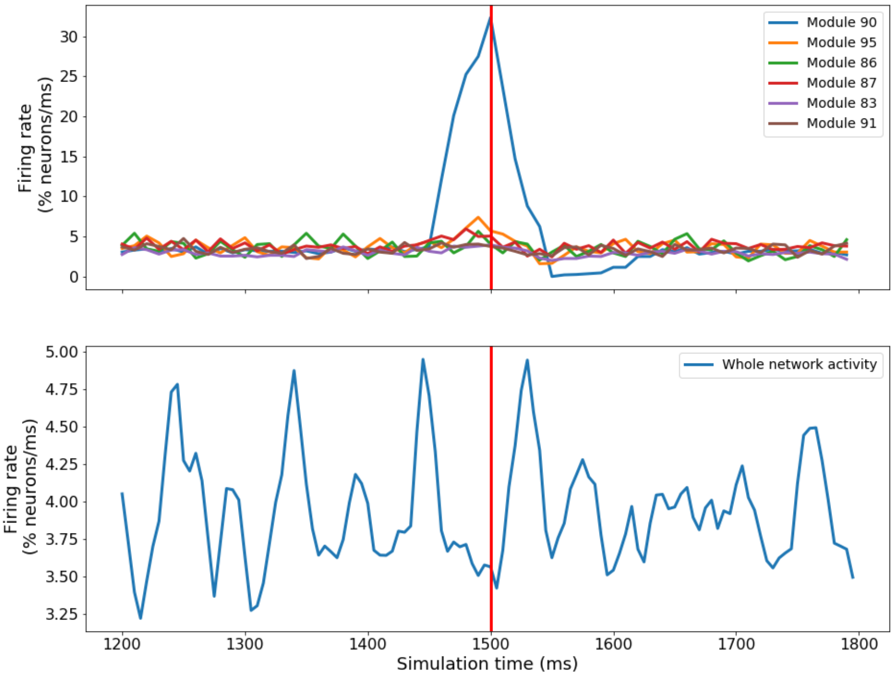


Figure 3.14: TMS stimulus effect on module 90 (receiver of the TMS puls) and the its most strongly connected neighbouring modules (top) and the whole network (bottom). The intermodular scaling is set to 50 in this simulation.

Table 3.4 shows the PCI values obtained from each of the 4 simulations corresponding to Figure 3.13. The PCI complexity between wakefulness and LoC simulations decrases, as in the original experiment. But once we compare the results with the same simulations without applying TMS, we see that the decreasing effect is the same regardless of whether or not we apply the pulse.

Discussion

The PCI values in Table 3.4 are in the range of 0.12 to 0.18. Compared to the PCI values obtained from human brains, which ranged from 0.2 to 0.6 approximately, our PCI values are very small. A recent study [20] has attempted to simulate PCI on a model with Kuramoto oscillators and their outcome was also somewhat insatisfactory. Their PCI values were low throughout their simulations. All in all, no satisfactory experiments reproducing the results of PCI on computational models exist as of this date.

Our hypothesis as to why the computational measurements differ so much from the clinical ones comes from the observation that there is a lack of significant and durable spatiotemporal response to TMS. Differently to how the human brain behaves, the computational models that we have worked with lack sustained activation after strong inputs are supplied focally to the model.

Perhaps, this is a consequence of the behaviour of oscillators. Oscillators have very regular activity and self-stimulation that makes them continue with the same patterns of group activation. This makes them immune to sudden changes in input as they can "absorb" its effect quickly and return to their normal behaviour.

Chapter 4

Conclusions

4.1 Challenges

I was very interested in the topic of computational neuroscience from the start, but had no experience in it. Thankfully, I had a fairly strong background on biology, but I had to learn everything on simulations of neural networks. This proved to be very difficult, especially because every simulation is coupled with some research and can't be guessed or made up. As a result, our work has involved many small experiments, each building up on top of another, as we figured out the best way to move towards large-scale models of the brain.

In terms of reproducing the PCI paper and other NCCs, the lack of existing research that has successfully done this has meant that our work has been largely original, which adds to the difficulty of our work.

4.2 Contributions

We set out to develop a thalamocortical brain model with realistic human-like dynamics that behaved according to some NCCs during loss of consciousness. Ambitiously, we did not limit ourselves to a single NCC, but instead aimed to test our experiment with as many as we could find and implement. The main outcome has been a detailed computational model of the brain that satisfies multiple NCCs, which we summarize below:

Our modular connectome experiment (Section 3.3.4) reproduces the effects of *cortical synchronization* representative of the transition from wakefulness to sleep, followed by a change from low-amplitude high-frequency oscillations to large-amplitude low-frequency *delta waves* (close to Theta band).

The thalamocortical model (Section 3.3.5) successfully demonstrates the expected decrease in *LZ complexity* of its activity as the input from the thalamus becomes stronger and induces system-wide synchronization, again, similar to what occurs during sleep.

The *PCI* experiment (Section 3.3.6), which involved a model of transcranial magnetic stimulation (TMS), was unsuccessful due to the inability of our TMS model to have a sustained effect on the network. The original PCI paper performed TMS on a human brain and recorded the spatiotemporal effect of this stimulation through EEG recordings. The human brain responds to TMS in a sustained fashion, allowing the stimuli to reverberate over time and results in a complex spatiotemporal response pattern. Our model did not behave this way, which means that PCI cannot be successfully measured. Retaining the stimulus information for longer periods of time may be a key feature of the human brain that allows it to create memories and other long-lasting patterns of activity. As discussed in the relevant section, other investigations have led to similar results regarding PCI on computational models, so some different approach is probably required to reproduce these results.

The previously mentioned contributions, in fact, build upon a number of other experiments and computational models that we have had to construct along the way. These intermediate experiments constitute an important contribution to the field of computational neuroscience: we provide a detailed account of computational models of different scales and complexities, explaining from the ground up how neurons are modelled, networks are put together and how realistic human-like dynamics can be achieved. This paves the way for researchers from different disciplines, even without a strong computational background, to use these computational simulations as a foundation in their own research or to further develop the ideas we have explored.

The archive associated with this report contains all these experiments and some others. It also includes implementations for the moving average, power spectral density, LZ76 and PCI algorithms, as well as Jupyter notebooks where the experiments were performed and the results analysed.

4.3 Future work

As a consequence of our findings during the TMS experiment, we pose the problem of understanding why a sustained reaction could not be achieved with the kind of computational models used. The richness of the spatial response seen in the brain is hardly reproduced by our models. It is important for this field of research to continue innovating and overcoming the limitations that arise over time. Given this limitation that we have found, it would be interesting to continue research in this direction and attempt to find answers to this problem. Perhaps it is the modular structure of our model, the scale of the simulation, non-realistic delays and weights, the presence of strong oscillatory dynamics, the neuron models or some other property of our models.

An aspect which we would have liked to improve is the thalamus model, which we oversimplified, lacking the time to invest on developing a more realistic model. It would be interesting to work on modelling the thalamus properly and simulate sleep through the hyperpolarization of thalamocortical cells, as really occurs in the brain.

Even though TMS did not properly work, we should work on improving our TMS model and perhaps employ a more rigorous approach to finding significant activation sources in the PCI calculation.

Bibliography

[1] D. Abásolo, S. Simons, R. Morgado da Silva, G. Tononi, and V. V. Vyazovskiy. Lempel-ziv complexity of cortical activity during sleep and waking in rats. *Journal of neurophysiology*, 113(7):2742–2752, 2015.

[2] M. Alkire, R. Haier, and J. Fallon. Toward a unified theory of narcosis: brain imaging evidence for a thalamocortical switch as the neurophysiologic basis of anesthetic-induced unconsciousness. *Consciousness and cognition*, 9(3):370–386, 2000.

[3] J. M. Amigó, J. Szczepański, E. Wajnryb, and M. V. Sanchez-Vives. Estimating the entropy rate of spike trains via lempel-ziv complexity. *Neural Computation*, 16(4):717–736, 2004.

[4] X. D. Arsiwalla, R. Zucca, A. Betella, E. Martinez, D. Dalmazzo, P. Omedas, G. Deco, and P. F. Verschure. Network dynamics with brainx3: a large-scale simulation of the human brain network with real-time interaction. *Frontiers in neuroinformatics*, 9:2, 2015.

[5] D. Bhowmik and M. Shanahan. Metastability and inter-band frequency modulation in networks of oscillating spiking neuron populations. *PloS one*, 8(4):e62234, 2013.

[6] C. G. Boeree. The emotional nervous system. `https://webspace.ship.edu/cgboer/limbicsystem.html`, May 2018.

[7] G. Buzsáki and A. Draguhn. Neuronal oscillations in cortical networks. *science*, 304(5679):1926–1929, 2004.

[8] A. G. Casali, S. Casarotto, M. Rosanova, M. Mariotti, and M. Massimini. General indices to characterize the electrical response of the cerebral cortex to tms. *Neuroimage*, 49(2):1459–1468, 2010.

[9] A. G. Casali, O. Gosseries, M. Rosanova, M. Boly, S. Sarasso, K. R. Casali, S. Casarotto, M.-A. Bruno, S. Laureys, G. Tononi, et al. A

theoretically based index of consciousness independent of sensory processing and behavior. *Science translational medicine*, 5(198):198ra105–198ra105, 2013.

[10] D. J. Chalmers. *The conscious mind: In search of a fundamental theory.* Oxford University Press, 1996.

[11] F. Crick and C. Koch. Towards a neurobiological theory of consciousness. In *Seminars in the Neurosciences*, volume 2, pages 263–275. Saunders Scientific Publications, 1990.

[12] F. Crick and C. Koch. A framework for consciousness. *Nature neuroscience*, 6(2):119–126, 2003.

[13] C. Eliasmith, T. C. Stewart, X. Choo, T. Bekolay, T. DeWolf, Y. Tang, and D. Rasmussen. A large-scale model of the functioning brain. *science*, 338(6111):1202–1205, 2012.

[14] S. K. Esser, S. L. Hill, and G. Tononi. Modeling the effects of transcranial magnetic stimulation on cortical circuits. *Journal of neurophysiology*, 94(1):622–639, 2005.

[15] S. K. Esser, S. L. Hill, and G. Tononi. Sleep homeostasis and cortical synchronization: I. modeling the effects of synaptic strength on sleep slow waves. *Sleep*, 30(12):1617–1630, 2007.

[16] A. Fernández, P. Zuluaga, D. Abásolo, C. Gómez, A. Serra, M. A. Méndez, and R. Hornero. Brain oscillatory complexity across the life span. *Clinical neurophysiology*, 123(11):2154–2162, 2012.

[17] W. Gerstner, W. M. Kistler, R. Naud, and L. Paninski. *Neuronal dynamics: From single neurons to networks and models of cognition.* Cambridge University Press, 2014.

[18] P. Hagmann, L. Cammoun, X. Gigandet, R. Meuli, C. J. Honey, V. J. Wedeen, and O. Sporns. Mapping the structural core of human cerebral cortex. *PLoS biology*, 6(7):e159, 2008.

[19] M. Hallett. Transcranial magnetic stimulation and the human brain. *Nature*, 406(6792):147–150, 2000.

[20] A. J. Ibáñez-Molina and S. Iglesias-Parro. A comparison between theoretical and experimental measures of consciousness as integrated information in an anatomically based network of coupled oscillators. *Complexity*, 2018, 2018.

[21] E. M. Izhikevich. Simulations of large-scale brain models. https://www.izhikevich.org/human_brain_simulation/Blue_Brain.htm#SimulationofLarge-ScaleBrainModels. Accessed: 2018-01-18.

[22] E. M. Izhikevich. Simple model of spiking neurons. *IEEE Transactions on neural networks*, 14(6):1569–1572, 2003.

[23] E. M. Izhikevich and G. M. Edelman. Large-scale model of mammalian thalamocortical systems. *Proceedings of the national academy of sciences*, 105(9):3593–3598, 2008.

[24] F. Kaspar and H. Schuster. Easily calculable measure for the complexity of spatiotemporal patterns. *Physical Review A*, 36(2):842, 1987.

[25] A. Lempel and J. Ziv. On the complexity of finite sequences. *IEEE Transactions on information theory*, 22(1):75–81, 1976.

[26] A. W. L. G. Matt Wimsatt, Jack Simpson. 3d brain. http://www.brainfacts.org/3d-brain, May 2018.

[27] D. Mumford. On the computational architecture of the neocortex. *Biological cybernetics*, 65(2):135–145, 1991.

[28] M. R. Rosenzweig, S. M. Breedlove, and A. L. Leiman. *Biological psychology: An introduction to behavioral, cognitive, and clinical neuroscience.* Sinauer Associates, 2002.

[29] M. Schartner, A. Seth, Q. Noirhomme, M. Boly, M.-A. Bruno, S. Laureys, and A. Barrett. Complexity of multi-dimensional spontaneous eeg decreases during propofol induced general anaesthesia. *PloS one*, 10(8): e0133532, 2015.

[30] Scholarpedia. Axonal conduction delays. http://www.scholarpedia.org/article/Axonal_conduction_delays, June 2018.

[31] A. K. Seth. The real problem. https://aeon.co/essays/the-hard-problem-of-consciousness-is-a-distraction-from-the-real-one. Accessed: 2018-01-23.

[32] A. K. Seth. Your brain hallucinates your conscious reality. https://www.ted.com/talks/anil_seth_how_your_brain_hallucinates_your_conscious_reality/transcript?language=en, May 2018.

[33] A. K. Seth, E. Izhikevich, G. N. Reeke, and G. M. Edelman. Theories and measures of consciousness: an extended framework. *Proceedings of the National Academy of Sciences*, 103(28):10799–10804, 2006.

[34] M. Steriade. Sleep oscillations and their blockage by activating systems. *Journal of Psychiatry and Neuroscience*, 19(5):354, 1994.

[35] M. Steriade, D. A. McCormick, and T. J. Sejnowski. Thalamocortical oscillations in the sleeping and aroused brain. *Science*, 262(5134):679–685, 1993.

[36] S. Studio. Brain structures and their functions. `http://serendip.brynmawr.edu/bb/kinser/Structure1.html`, May 2018.

[37] G. Tononi. An information integration theory of consciousness. *BMC neuroscience*, 5(1):42, 2004.

[38] G. Tononi and C. Koch. The neural correlates of consciousness. *Annals of the New York Academy of Sciences*, 1124(1):239–261, 2008.

[39] Wikipedia. Mann-whitney u test. `https://en.wikipedia.org/wiki/Mann%E2%80%93Whitney_U_test`, June 2018.