A theory of how the brain computes

Tan, Teck Liang

2018

Tan, T. L. (2018). A theory of how the brain computes. Doctoral thesis, Nanyang Technological University, Singapore.

<http://hdl.handle.net/10356/73797>

<https://doi.org/10.32657/10356/73797>

A Theory of How The Brain Computes

by

TAN Teck Liang

A thesis submitted to the Nanyang Technological University
in partial fulfillment of the requirement for the degree of
Doctor of Philosophy

in the
School of Physical and Mathematical Sciences
Division of Physics and Applied Physics

2017

Acknowledgements

In this culmination of my 4-year research program, I am especially grateful to my doctoral advisor Assoc. Prof. Cheong Siew Ann for his mentorship, encouragement, and wisdom that made this thesis and the path to graduation possible. I am thankful also to the members of my Thesis Advisory Committee, Assoc. Prof. Chew Lock Yue and Asst. Prof. Xu Hong, for their guidance and suggestions over the years. I am extremely fortunate to have my friends and colleagues Darrell, Boon Kin, Woon Peng, Wenyan, James, Chee Yong, and Misha from the Statistical and Multi-scale Simulation Laboratory with whom I enjoyed endless hours of productive discussions and fun times.

I thank also the School of Physical and Mathematical Sciences for their generous scholarship that funded this research.

Contents

Acknowledgements	i
List of Figures	iv
List of Tables	v
Abstract	vi
1 Introduction	1
1.1 Chasing the ever elusive brain research	1
1.2 Edge of Chaos, Criticality and the Brain	3
1.3 Organisation of Thesis	5
2 Literature Review	6
2.1 A Short History of Computing	6
2.1.1 Representation, Transformation and Logical Operations . . .	7
2.1.2 Long- and Short-Term Memory	7
2.2 Nonlinear Dynamics and Chaos	8
2.2.1 Neuron Model	9
2.2.2 Integrate-and-fire	9
2.2.3 Hodgkin-Huxley	9
2.2.4 Izhikevich's Simplified Neuron Model	11
2.3 Neuroscience and Complex Networks	14
2.4 Information Processing and Synchronisation	15
2.5 The Flow of Information and Current Research Gap	16
3 Statistical Complexity is Maximized in a Small-World Brain	17
3.1 Introduction	17
3.2 Izhikevich's Simplified Neuron Model	19
3.2.1 Phase Diagram	20
3.3 Parametric Coupling	24
3.3.1 Phase-Diagram-Inspired Energy Landscape	26
3.3.2 Quasi-Equilibrium Distribution of Parameters	28
3.4 Complexity and Small-World Network	28

3.4.1	Complexity Measures	30
3.4.2	Maximising of Complexity	32
3.5	Chapter Conclusions	34
4	Robust Clustering Analysis of Synchronisation in Small-world Brain Networks	36
4.1	Pairwise ISI-distance	37
4.1.1	Sparse Representation of Times Series	38
4.2	Robust Hierarchical Clustering	39
4.3	Sliding Windows Analysis	42
4.4	Information Flow Visualisation: Alluvial Diagram	43
4.4.1	Tracing a Stream of Robust Clusters	49
4.4.2	Input Signal	51
4.5	A Quantitative Approach to Validate: Adjusted Rand Index	53
4.6	Chapter Conclusions	57
4.7	MATLAB function for Alluvial Diagram	58
4.8	Other Sorted Alluvial Diagrams for a Izhikevich Neuron Network	61
5	Conclusion	64
	Bibliography	68

List of Figures

1.1	The bifurcation diagram of the logistic map	4
2.1	Basic components of Hodgkin-Huxley type models	10
2.2	Izhikevich's simple model	12
2.3	A summary of parameters used in Izhikevich's model	13
3.1	Featured parameters used	19
3.2	Simulated Time series of the Izhikevichs neuron model	20
3.3	Observable differences in the time series	22
3.4	Phase diagram for parameters c and d	23
3.5	Phase Diagram Inspired Energy Landscape	27
3.6	Frequency distribution of parameters	29
3.7	Effects of rewiring probability	30
3.8	Illustration on Statistical Complexity	32
3.9	Maximizing of Statistical Complexity in Neuron Dynamics with Small-world Property	33
4.1	Sparse Representation of Times Series	38
4.2	Hierarchical Clustering and it's Dendrogram	40
4.3	Hierarchical Clustering and it's Robust Length	41
4.4	Sliding Windows Analysis	43
4.5	Alluvial Diagrams illustrating Structural Change	46
4.6	Example on Alluvial Diagram	48
4.7	Alluvial Diagram for a Izhikevich Neuron Network	49
4.8	Sorted Alluvial Diagram for a Izhikevich Neuron Network	50
4.9	Sorted Alluvial Diagram for a Controlled Izhikevich Neuron Network	52
4.10	Adjusted Rand Index of Resting State	56
4.11	Adjusted Rand Index of Controlled State	56
4.12	Other Sorted Alluvial Diagrams for a Izhikevich Neuron Network	61
4.13	Other Sorted Alluvial Diagrams for a Izhikevich Neuron Network	62
4.14	Other Sorted Alluvial Diagrams for a Izhikevich Neuron Network	63
5.1	Thesis Milestones	66

List of Tables

4.1 Alluvial Diagram Example	44

Abstract

We study a network of Izhikevich neurons to explore what it means for a brain to be at criticality. We first constructed the phase diagram of a single Izhikevich excitatory neuron, and identified a small region of the parameter space where we find a large number of phases to serve as our edge of chaos. We then couple the outputs of these neurons directly to the parameters of other neurons, so that the neuron dynamics can drive transitions from one phase to another on an artificial energy landscape. We measure the statistical complexity of the parameter time series, while the network is tuned from a regular network to a random network using the Watts-Strogatz rewiring algorithm. We find that the statistical complexity of the parameter dynamics is maximized when the neuron network is most small-world-like. We also study the ability of Izhikevich neurons to synchronise and the conditions under which such synchronisation occurs. We then implored the robust hierarchical clustering technique with sliding window analysis based on interspike-intervals (ISI) distance to find the synchronization clusters of neurons their evolution through over time in the form of an alluvial diagram. We seek to gain insights into how a neuronal network processes information from this method.

Chapter 1

Introduction

1.1 Chasing the ever elusive brain research

Since I first began working on projects relating to this thesis back in August 2013, much has changed at the frontiers of artificial intelligence. The earliest prominent use of personal computers were much like the industrial revolutions in machines: to do dumb things but at an amazing speed. The game changed in the 1990s with the discovery of backpropagation algorithm [1] and popularisation of artificial neural networks (ANNs) [2–4] which is said to be inspired by the working principles of a biological neural network. While it may seem that with cognitive computer programs and the endless scaling potential of computing power known as the Moore’s law, we would quickly run out of problems to solve, but it turned out not to be the case. At least not without limits.

I had the honour of attending a talk from a scientist I revered as one of the giants of complexity science, John Henry Holland (February 2, 1929 - August 9, 2015) speaking at East-West Connections - “Cultural Circuits in the Human Brain” (15 - 16 September 2014, Singapore). In this talk, he said: “even with the technology of artificial intelligence today, a person, with a few days of practices, can easily defeat a computer opponent in a game of Go”. The seemingly infinite number of possibilities with the given set of rules in the game are just too vast for traditional

computer algorithms based on exhaustive search to process, yet the human brain is able to do using less time and at a lower cost (in terms of energy consumption). However, this is no longer the case today. AlphaGO, an AI computer program developed by DeepMind, plays the board game Go and shocked the world by beating a human professional Go player without handicaps on a full-sized 19×19 board in October 2015 [5, 6]. With a combination of machine learning and tree search techniques, together with extensive training from both human and computer play, AlphaGo's algorithm popularised artificial neural networks again, but this time a different aspect of it: deep learning.

While we celebrate all these achievements, is deep learning the answer to the question of how the brain works? The answer to that is simply no. This is because the brain operates on a very different scale. Robert M. Metcalfe, the director of SiCortex scientific supercomputer firm, mentioned that the brain, working at around 20 watts, is 10^4 times faster than that of SC5832 super computer working at 20 kilowatts [7]. Jeff Hawkins offers an explanation to the vast difference in computations speeds - memory. Jeff points out the fallacy of comparing a brain to a computer through his thought experiment called 'one hundred-step rule' where he argued that the time taken from the light entering the eye to the reaction of the person after recognizing the image involves "a chain no longer than one hundred neurons", whereas "a digital computer attempting to solve the same problem would take billions of steps" [8].

That said, the debate today is still far from being done and dusted. We believe the thought experiment presented by Hawkins failed to capture the learning and training a human brain received to be able to perform highly complicated task(s). Given the right training and rewiring at the structural level, a computer, in principle, can shave away the millions or even billions of redundant repetitive steps because, after all, memory is not an exclusive technology of the brain.

1.2 Edge of Chaos, Criticality and the Brain

Switching perspective to the natural sciences may offer yet another insight to how the brain operates: the edge of chaos, a phenomenon observed in dynamical systems. Stuart A. Kauffman, another giant of complexity science, used this idea in the late 1980s to early 1990s to explain many biological-like systems that can be seen in many of his works tackling issues on self-organization and far-from-equilibrium dynamics explaining the complexity of biological systems [9–12]. Together with Christopher Langton and Norman Packard, Kauffman advocated that “life exist at the edge of chaos” [13–15]. This leads to the initial suspicion that the edge of chaos is the answer that we are looking for in the brain. However, present consensus among scientists have since moved from the notion of edge of chaos to the use of criticality and more specifically self-organised criticality [16–19]. That said, naturally, the next step is to ask the following three key questions: ① what does it mean for the brain to be at criticality? ② Why is the brain at criticality? And lastly ③ how does the brain get to criticality?

To answer these questions, let me start with logistic map:

$$x_{n+1} = rx_n(1 - x_n) \quad (1.1)$$

and its bifurcation diagram as an example (Fig. 1.1). As the parameter r is increased, the logistic equation goes through the period-doubling approach to chaos, i.e. from one fixed point to the 2-cycle, and from the 2-cycle to the 4-cycle and so on until we reach a point $r \approx 3.5699$, the edge of chaos where any further beyond there is chaos. Between the ordered regime with few periodic cycles and the disordered regime with chaos, we have a very narrow range of r where we can have large finite number of periodic cycles. While this example is simple to illustrate and connections had been drawn that it exhibits behaviours that are universal to other systems at criticality: fractal geometry and power-law distributions [20–22], is it however, contested [18]. Nonetheless, we interpret that the brain is at criticality would meant that the brain exhibit properties that are power law distributed.

FIGURE 1.1: The bifurcation diagram of the logistic map, $x_{n+1} = rx_n(1 - x_n)$. The bifurcation parameter r is shown on the horizontal axis of the plot and the vertical axis shows the fixed points and period cycles of the logistic map.

Secondly, we believe the reason why the brain is at criticality is so that it is the most effective computationally [23, 24]. Being at criticality clearly allows small changes to the control parameter to bring access to rich variation in the meaningful and observable ways that the brain goes about processing information and performing computation.

Lastly, the question on how the brain gets to criticality would lead us to the key research questions in this thesis research. We would like to know how is the encoding of information is done, and thereafter how information is transformed. Further possible research questions like what are the mechanisms and processes involved as the brain go through evolution (animal versus human), development (baby versus adults), and modulation (external effects like drugs).

1.3 Organisation of Thesis

The thesis is organized into five chapters with this being the first.

In Chapter 2, we review the existing literature relevant to the construction and understanding of neuron models, topology of neuron networks and synchronisation of neuronal activities. We will outline the frontier of the existing works and how our thesis builds upon them.

In Chapter 3, we study a network of Izhikevich neurons to explore what it means for a brain to be at criticality. To do so, we first constructed the phase diagram of a single Izhikevich excitatory neuron, and identified a small region of the parameter space where we find a large number of phases to serve as our edge of chaos. We then couple the outputs of these neurons directly to the parameters of other neurons, so that the neuron dynamics can drive transitions from one phase to another on an artificial energy landscape. Finally, we measure the statistical complexity of the parameter time series, while the network is tuned from a regular network to a random network using the Watts-Strogatz rewiring algorithm. We find that the statistical complexity of the parameter dynamics is maximized when the neuron network is most small-world-like.

In Chapter 4, we study the ability of Izhikevich neurons to synchronise and the conditions under which such synchronisation occurs using a sparse representation of the neuron's voltage time series and a pairwise interspike-intervals distance measure. We then used the robust hierarchical clustering technique with sliding window analysis to extract the groups of neurons that are synchronised at different times. Lastly, we implemented a visualisation tool based on alluvial diagrams in MATLAB to observe the robust synchronised clusters and their evolutions through different time windows of the simulations.

In Chapter 5, we outline the contributions this thesis has made and conclude.

Chapter 2

Literature Review

2.1 A Short History of Computing

Interestingly, the earliest usage of the word ‘*computer*’ referred to an actual human person who performed calculations and not the electronic device that we are familiar with. That aside, according to the book *A Brief History of Computing* by Gerard O’Regan, analog computers were introduced around early nineteenth century by James Thompson and the first large-scale general-purpose mechanical analog computer was developed in the late 1920s by Vannevar Bush and others at the Massachusetts Institute of Technology [25, 26].

The digital computer came into existence only after the 1940s [25], with binary information being stored in vacuum-tube transistors. Vacuum-tube transistors were later replaced by solid-state transistors invented by Shockley and others in the 1950s [25, 27]. The difference in the information representation between analog and digital computers was the main reason that led to the dominance of the latter.

2.1.1 Representation, Transformation and Logical Operations

Representation of information in an analog computer, can be for example, a single capacitor storing one continuous variable based on the physical quantity such as the voltage, whereas a digital computer used several logical units (vacuum tubes in the early days and later transistors) and each storing only one binary value ‘0’ or ‘1’ [25]. The trade off is the accuracy of the data where analog computers are limited by analog noise while digital computers have quantization noise [25].

After understanding how information is represented, the next question we ask is how the information can be processed and transformed? Analog computers performed operations in parallel [25] while digital computer performed logical operations one bit at a time. The basic architecture for doing so has remained largely unchanged since the 1940s, following the van Neuman architecture [28].

Because digital computers are so common, we may be tempted to think of information representation in the brain in digital terms.

2.1.2 Long- and Short-Term Memory

Understanding long- and short-term memory in today’s computers is relatively simple: modern long-term memory are those data stored in hard disks drives (HDD), solid-state drives (SDD), etc. On the other hand, short-term memory are best understood referring to the random access memory (RAM).

However, this distinction of hardware-to-functionality was not as clear-cut when it comes to the brain in the beginning. The psychology discipline in the 1970s had very intensive debates over the modeling and definition of long-term-memory (LTM) and short-term-memory (STM) and their functionality in learning [29–31]. The debate later shifted focus to the processes involved in memory starting from encoding to sorting and recalling [32, 33]. Many years later, neuroscientists, using functional imaging of brain responses [34–36], mapped out the neuroanatomy

of memory [36–38]. The general consensus is that the hippocampus is mainly responsible for LTM while the prefrontal cortex is responsible for STM. There is also another class of memory known as the skill memory, which keeps the automatic learned memories like riding a bicycle, and is processed in the cerebellum [39, 40].

The formation of LTM is better understood compared to STM because it involves the strengthening of certain synaptic connections and weakening of others. These changes are permanent, and can be captured through functional imaging techniques [41, 42]. The formation of STM however is more complicated as patterns observed are transient and imaging techniques are still incapable of resolving individual neurons. This raises the next question: apart from knowing where the brain stores memory, how can we find out more about the mechanisms involved?

This ignorance notwithstanding, we understand that, albeit having limited imaging resolution, the data showed the lighting up of a bunch of neurons around the prefrontal cortex whenever the the subject is performing activities linked with STM. This suggested a certain level of synchronization at least within the region reported because of the persistent and yet non-permanent signals. Such phenomena of localized synchronization is also often observed in nonlinear systems by physicists.

2.2 Nonlinear Dynamics and Chaos

In the middle of the 20th century, the study of chaos and nonlinear dynamics blossomed. Details of how chaos arises from nonlinear dynamics will not be discussed here, but let us point out that the concept of controlling chaos emerged in the early 1990s and drew much attention from disciplines outside of physics [43, 44]. Methods to control chaos includes parameter variation, entrainment and migration controls, engineering control, and synchronization [44].

As mentioned earlier, in this thesis the synchronization of nonlinear (or perhaps even chaotic) systems are of particular interest. Mirollo and Strogatz reported

the synchronization of pulse-coupled biological oscillators [45] in 1990, where they used an integrate-and-fire model based on Peskin’s model of the cardiac pacemaker. They were successful in demonstrating the emergence of the synchrony both numerically and through some theoretical discussion. However, their direction of research focus shifted in the 1990s into the mathematical analysis of the bifurcation in coupled oscillators [46] which is of less relevance to this thesis. It was not until the early 2000s where we find further applications of nonlinear dynamics analysis in neuroscience related field [47–50]. This development came a puzzling half a century after the first major breakthrough in neuron modeling, by Hodgkin and Huxley in 1952. We will therefore detour a little to talk about neuron models in the next section before coming back to visit further works on synchronization and the brain in the last section of this chapter.

2.2.1 Neuron Model

2.2.2 Integrate-and-fire

One of the earliest known biological neuron model dates back to 1907 when Lapicque introduced the integrate-and-fire model even before the mechanisms responsible for the generation of neuronal action potentials were known [51]. The variant ‘leaky integrate-and-fire’ model was developed later and is vastly popular in the theoretical study of the dynamics in spiking neural networks for its computational efficiency [52–55]. Another variant known as the ‘exponential integrate-and-fire’ model was developed by Romain Brette and Wulfram Gerstner [56], and this showed promise in the prediction of spike timings.

2.2.3 Hodgkin-Huxley

The major breakthrough mentioned earlier was the model by Hodgkin and Huxley in 1952, where they were able to include the dynamics of the voltage-dependent membrane conductances responsible for the observed action potentials [57].

FIGURE 2.1: Basic components of Hodgkin-Huxley type models. Hodgkin-Huxley type models represent the biophysical characteristic of cell membranes. The lipid bilayer is represented as a capacitance (C_m). Voltage-gated and leak ion channels are represented by nonlinear (g_n) and linear (g_L) conductances, respectively. The electrochemical gradients driving the flow of ions are represented by the respective batteries (E_n and E_L), and ion pumps and exchangers are represented by current sources (I_p). (Figure and caption obtained from Wikipedia)

Mathematically, the voltage-current relationship flowing through the lipid bilayer is given by

$$I_c = C_m \frac{dV_m}{dt}. \quad (2.1)$$

The current through a given ion channel ($i = \{\text{Na}, \text{K}, \text{l}\}$ representing sodium, potassium, and leakages respectively) is governed by Ohm's Law as

$$I_i = g_i(V_m - V_i), \quad (2.2)$$

where V_i is the reversal potential of the i -th ion channel. Hence, the total current through the membrane, $I = I_c + I_i$, is given by

$$I = C_m \frac{dV_m}{dt} + g_K(V_m - V_K) + g_{Na}(V_m - V_{Na}) + g_l(V_m - V_l). \quad (2.3)$$

Since the first major breakthrough in development of neural model by Hodgkin

and Huxley in 1952, numerous biophysically detailed Hodgkin-Huxley-type neural models have been developed [58–62]. All of these models were created with two main aims, firstly be physically accurate and secondly be computationally efficient.

2.2.4 Izhikevich’s Simplified Neuron Model

Izhikevich demonstrated that his model was able to achieve both “biologically plausibility of Hodgkin-Huxley-type dynamics and the computational efficiency of integrate-and-fire neurons” [62]. He developed bifurcation methodologies [63] which enabled him to reduce the biophysically accurate Hodgkin-Huxley-type neuronal models to a two-dimensional (2-D) system of ordinary differential equations of the form

$$\dot{v} = 0.04v^2 + 5v + 140 - u + I, \quad (2.4)$$

$$\dot{u} = a(bv - u), \quad (2.5)$$

with the auxiliary after-spike resetting

$$\text{if } v \geq 30\text{mV, then } \begin{cases} v \leftarrow c \\ u \leftarrow u + d. \end{cases} \quad (2.6)$$

Here, a , b , c , and d are dimensionless parameters, and the notation $\dot{v} = dv/dt$, where t is the time. The variables v and u are dimensionless and represents the membrane potential of the neuron and membrane recovery variable respectively. The membrane recovery variable, u , provides negative feedback to v and accounts for the activation of K^+ ionic currents and inactivation of Na^+ ionic currents. The auxiliary after-spike reset activates after the spike reaches its peak (+30 mV) and according to Eq. (2.6). Lastly, I represents the synaptic currents or injected dc-currents.

The numerical values in Eq. (2.4) were obtained by fitting the spike initiation dynamics of a cortical neuron to allow the membrane potential v having mV scale and the time t in ms scale [62]. The parameter a describes the time scale of the

FIGURE 2.2: Izhikevich's simple model. Eq. (2.4) and Eq. (2.5) can reproduce firing patterns of neurons recorded from the rats motor cortex. Model parameters as in Fig. 2.3. (Figure obtained from Ref. [62])

recovery variable u , while b describes the sensitivity of the recovery variable u to the sub-threshold fluctuations of the membrane potential v . The parameter c describes the after-spike reset value of the membrane potential v caused by the fast high-threshold K^+ conductances while d describes after-spike reset of the recovery variable u caused by slow high-threshold Na^+ and K^+ conductances. Through variation of the parameters, various intrinsic firing patterns, including those exhibited by the known types of neocortical [64–66] and thalamic neurons are obtained (see Fig. 2.3). He was able to demonstrate collective behaviors and rhythms similar to that of the mammalian cortex with his pulse-coupled neural network [62].

One example of such application of complex network done by Izhikevich and Edelman extending the simple model to stimulate in large scale in Ref. [67], and they

FIGURE 2.3: A summary of parameters used in Izhikevich's model. Known types of neurons corresponding to different values of the parameters a , b , c , and d in the model described by Eq. (2.4) to Eq. (2.6). RS, IB, and CH are cortical excitatory neurons. FS and LTS are cortical inhibitory interneurons. Each inset shows a voltage response of the model neuron to a step of dc-current $I = 10$ (bottom). Time resolution is 0.1 ms. (Figure and caption obtained from Ref. [62])

used macroscopic anatomy from diffusion tensor imaging (DTI) data derived from magnetic resonance imaging (MRI). The microcircuitry Izhikevich and Edelman used was data based on detailed reconstruction studies of the cat primary visual cortex by Binzegger *et al.* [68]. Given the success of this model, we decided to adopt the model in constructing the brain network to study the brain at the edge of chaos. Right now, there is still one more missing gap to fill, and that is how much do we actually know about our brain today?

2.3 Neuroscience and Complex Networks

To understand something in science, we have to observe the behavior of a subject, and if possible, perform experiments on it. Since the experiments on a living human is taboo, in neuroscience we are severely limited when it comes to direct observations of the brain. Indirect observation methods developed to overcome such limitations are known as functional imaging techniques. Such imaging techniques include functional magnetic resonance imaging (fMRI), positron emission tomography (PET), and electroencephalography / magnetoencephalography (EEG/MEG). A notable initiative in this field is the Human Connectome Project which “aims to characterize human brain connectivity and function in a population of 1200 healthy adults and to enable detailed comparisons between brain circuits, behavior, and genetics at the level of individual subjects” [69].

Going more in-depth, one would find the early foundations of mapping connectivity in resting-state human brain using fMRI were laid by Biswal *et al.* and Lowe *et al.* through finding consistent temporal correlations [70, 71] while Shulman *et al.* and Raichle *et al.* working with PET drew similar conclusions [72, 73]. These resting-state research have spurred great scientific interest searching for a baseline state in the human brain [74–85]. These series of research went from studying localized areas of interest to connectivity in large-scale networks.

Apart from advancements from the neuroimaging community, the complexity science community has also become increasingly interested in the high-resolution data coming from functional imaging [86–90]. One such research area is the topology of the connection in brain cells [91]. Much work has been done on describing the connection using small-world networks [91–96]. The emergent of small-world networks found by independent research groups cannot be coincidental. The pursue of using network theory, and particularly of small-world networks, to decipher and understand the functional groups in the brain network will be one of the main themes in this thesis.

Last but not least, in vitro experiments were also performed. Notably, the work by John M. Beggs *et al.*, on retinal tissue, cortical slices and cultures [97–99]. They developed an initial model to study critical branching [97], neuronal avalanche [98], and spatial and temporal correlations [99]. While this thesis do not work on empirical data, it would be interesting to work with an experimentalist who can construct similar neuronal system to verify our model’s findings.

2.4 Information Processing and Synchronisation

Apart from the topological aspects of the brain, comprehending the mechanism of information processing is also a hotly debated challenge, and therefore we would like to review the signals processing methodologies employed thus far in neuroscience. Early studies built around the idea where the encoding of information is done using the spiking rates of the neurons [100–102]. Bialek *et al.* identified the weaknesses of these early methods and formulated strategies for characterizing the neural code from the point of view of the organism, culminating in algorithms for real-time stimulus reconstruction based on a single sample of the spike train [102]. This is because a single spike event may be responsible for the discrimination between different stimuli [103, 104], and thus a pure spiking rates analysis is insufficient.

Two large classes of mathematical tools have been developed over the years to tackle this problem, and they are namely ① phase comparison methods based on Hilbert transform or Fourier transform or any other variants [105, 106] while ② are distance comparison methods based on time intervals between spikes [103, 107–109]. Both classes of methods provide means to detect and measure the degree of neuronal synchrony - an indication of neurons working together [45, 110–112].

2.5 The Flow of Information and Current Research Gap

To bridge the gap between many fields studying an old problem studying the flow of information in the brain, we need to look broad and deep. Fred Dretske, a philosopher, attempted this by crossing the idea of information theory by Claude Shannon to develop “a theory of knowledge and a philosophy of mind” [113], while neuroscientist generally adopted a more evidence-based approach to map the neural network and attempt to describe the flow of information [114–126]. Our approach is to investigate how information can be encoded in the Izhikevich neuron network in terms of synchronization clusters and demonstrate the flow of information by tracing how the synchronization clusters evolve in such a network over time. This can be seen as an extension to the work done by Diesmann M et al. [127] on how neuron synchrony propagates in a along a fully connected synfire chain.

Chapter 3

Statistical Complexity is Maximized in a Small-World Brain

3.1 Introduction

The brain, often compared to the central processing unit (CPU) of a digital computer, handles and processes information from all our sensory organs. We clearly benefit from its capability to hold thoughts and consciousness, even though the working principles of the brain is not yet fully understood. While neuron level dynamics has been thoroughly studied [52], and extensive work has been done to map the functional regions [128], we do not have an established theory connecting functional regions to information processing at the whole-brain level. Unlike a digital computer, the brain is complex, and its information processing capabilities might be an emergent property [129]. Indeed, the brain is suggested to be critical — with power laws found, first by Bak *et al.* in their learning model [130], then by Beggs in neurons grown on petri dishes [131], and lastly by Kitzbichler *et al.* in vivo in a functioning human brain [132]. These discoveries prompted science journalist to claim that “the human brain is on the edge of chaos” [133, 134].

Chialvo further argued that the emergent complex neural dynamics in brains are essential to navigate a complex, critical world [129]. The logic behind the argument is that critical systems can display long-range (power-law) correlations in space and time, and therefore it is very important to allow neural networks that process information from such systems to have neuron-neuron interactions that go beyond immediate neighbours. Indeed, computer scientists have shown that the computational capability of a neural network is maximised at the edge of chaos [135, 136], so that it is able to assign unique non-random outputs to the most number of inputs. Crutchfield and Young earlier developed a measure of statistical complexity based on his ϵ -machines quantifying the information density of complex patterns [137, 138], and found that the complexity of a dynamical system is maximized at the edge of chaos. At the same time, others looking into brain-related networks discovered that they have small-world properties [91–93]. Is this network topology unrelated to the edge of chaos, or is it a consequence of the brain being at criticality? Our goal is to understand the brain more deeply by linking all these parallel concepts to ask broader questions as what it means for the brain to be at criticality? Why is it at criticality? How does it get to criticality? What can it do at criticality? Specifically in this Chapter, we check whether complexity is indeed maximized for a small-world brain.

We organised our Chapter into five sections. In Section 3.2, we explain the need to pick a biologically realistic neuron model. This is because we are interested in emergent phenomenon, therefore we should not use models at the functional or whole-brain level. In the literature, the Hodgkin-Huxley model [57] is the most realistic, but it is at the same time the most computationally expensive. On the other hand, integrate-and-fire models [51] are computationally cheap, but they are not realistic. We strike a compromise by using Izhikevich’s model [62], which balances realism with computational tractability. We then map out the detailed phase diagram of a single neuron which lead us to discover a narrow region of the phase diagram that is reminiscent of the edge of chaos. In Section 3.3, we test whether it is possible to have neurons ‘kick’ each other from one phase to another in this narrow region of the phase diagram, by parametrically coupling

neurons together in an artificial energy landscape. We found that the neurons eventually relaxed to a distribution consistent with equilibrium thermodynamics. In Section 3.4, we measure the transient dynamics of various small-world networks of neurons, and found that the measured statistical complexity of their dynamics peaks close to where the network is most small-world-like, as characterised by the gap between the average clustering coefficient and the average path length.

3.2 Izhikevich's Simplified Neuron Model

Integrate-and-fire models [51] introduced as early as 1907 and their variants [54, 55] were the first ever models used to mimic the dynamics of a neuron. These models were highly popular at first as they capture the essence of neuron firing behaviour without detailed consideration of the biological mechanisms, and at a very low computational cost. On the other hand, the Hodgkin-Huxley model [57], introduced in 1952, was hailed as the first biologically-inspired model that captures not only the overall dynamics of the neuron but also detailed the internal mechanism. Several other biologically-inspired models were developed subsequently [139] aiming to reduce computational cost. We eventually chose the computationally lightweight Izhikevich's model in which different neuron types corresponds to different parameters (see Fig. 3.1 or Ref. [62]).

FIGURE 3.1: Featured parameters used to simulate seven different spiking types in Izhikevich's model (redrawn from Fig. 2 of [62]).

The summary of parameters used to simulate seven different spiking types in Izhikevich's model can be seen in Fig. 3.1 or Ref. [62]. The excitatory neurons are divided into three classes: RS (regular spiking), IB (intrinsically bursting), and CH (chattering), while the inhibitory neurons are divided into two classes: FS (fast spiking) and LTS (low-threshold spiking). There are two other neuron types that fall outside of the above classification scheme, and they are TC (thalamo-cortical) neurons and RZ (resonator) neurons.

FIGURE 3.2: Simulated time series of the Izhikevich's neuron model based on Eq. (2.4) to Eq. (2.6) and parameters provided in Fig. 3.1.

3.2.1 Phase Diagram

Even though Izhikevich developed his simplified model relying on the fact that the spiking behaviour changes abruptly when we go from one spiking type to another, he did not sketch a phase diagram for his model. Touboul performed a rigorous analysis on parameters I , a and b to identify the bifurcations separating

different spiking phases on the a - b parameter space [140], but not on c and d which appears in the reset conditions, Eq. (2.6). Hence, our first order of business is to fully characterize the phase diagram of a single neuron. To do this, we adopted the 4th-order Runge-Kutta method (RK4) with step size $h = 10^{-3}$ to implement Eq. (2.4) and Eq. (2.5) following the steps:

$$\begin{aligned}
A_1 &= 0.04v_n^2 + 5v_n + 140 - u_n + I, \\
B_1 &= a(bv_n - u_n), \\
A_2 &= 0.04(v_n + hA_1/2)^2 + 5(v_n + hA_1/2) + 140 - (u_n + hB_1/2) + I, \\
B_2 &= a[b(v_n + hA_1/2) - (u_n + hB_1/2)], \\
A_3 &= 0.04(v_n + hA_2/2)^2 + 5(v_n + hA_2/2) + 140 - (u_n + hB_2/2) + I, \\
B_3 &= a[b(v_n + hA_2/2) - (u_n + hB_2/2)], \\
A_4 &= 0.04(v_n + hA_3/2)^2 + 5(v_n + hA_3/2) + 140 - (u_n + hB_3/2) + I, \\
B_4 &= a[b(v_n + hA_3/2) - (u_n + hB_3/2)], \\
v_{n+1} &= v_n + h(A_1 + 2A_2 + 2A_3 + A_4)/6, \\
u_{n+1} &= u_n + h(B_1 + 2B_2 + 2B_3 + B_4)/6.
\end{aligned} \tag{3.1}$$

The terms A_1 , A_2 , A_3 , A_4 and B_1 , B_2 , B_3 , B_4 are intermediates required to be computed for the variables v_n and u_n respectively. Here v_n and u_n are the membrane potential variable and the membrane recovery variable at time $t = nh$ respectively. Then, with the input of initial conditions v_0 and u_0 , together with the parameters h , I , a , b , c and d we can compute from Eq. (3.1) the values of v_n and u_n .

FIGURE 3.3: Observable differences in the time series with gradual variation of the parameters c and d from IB to CH. The nomenclature here follows the number of spikes in the spike trains observed and the Roman numbering is assigned respectively.

After replicating the results from Izhikevich’s model, further exploration revealed distinct changes in the time series, seen in Fig. 3.3, as the spiking type goes from IB to CH as we vary the parameter in the c - d plane. A systematic examination of Izhikevich’s model in the c - d plane yields the phase diagram shown in Fig. 3.4. In particular, the “rainbow” CH regime is reminiscent of the period doubling route to chaos in the logistic map, and is thus a very promising area to explore for possible encoding of information, as each “shade” enclosed one distinct spiking time series. We can treat the small region in parameter space with a proliferation of regimes before the system enters the FS regime as the edge of chaos and these different spiking regimes as the basis to build a computational representation of information.

FIGURE 3.4: Phase diagram for parameters c and d while fixing $a = 0.02$ and $b = 0.2$, which is the common parameter values for RS, IB, and CH seen in Ref. [62]. This plot is obtained at integrating step size of $h = 5 \times 10^{-3}$, and resolution of parameters c and d is 2×10^{-2} . Each shade of colour is assigned after systematical examination of the time series of the membrane potential, v , in accordance to that displayed in Fig. 3.3.

As we have explained in Chapter 1, we believe that the brain is at criticality in order to process complex information sensed from the environment. However, it is possible that the whole brain is at criticality, but none of its functional parts are. Alternatively, the brain can be at criticality at every scale. That is to say, even if we restrict ourselves to examining a small part of the brain, even this small part resides at criticality. We do not know which scenario is true, or whether the biological brain is somewhere between these two extremes. In fact, there is no literature on the frequency distribution of neuron types in a functional brain. In other words, if we imagine we can extract the Izhikevich parameters for each and every neuron in the brain, and plot them as points in the Izhikevich parameter space, we do not know whether these points are uniformly distributed (the whole

brain is at criticality, but not all its parts), or they are clustered (the brain is at criticality at multiple scales). Even though we do not have the answer to the above question, we felt we can still take a baby step forward to understanding the problem of information processing by a brain at criticality by studying a small network of CH neurons all of which are close to the edge of chaos [141–143] (and by extension, at criticality). If we can demonstrate that a small network of CH neurons can create a system at criticality to encode information, then a larger network using similar building blocks will then be able to do more.

3.3 Parametric Coupling

In the literature, neurons are mostly pulse-coupled [144]. Some papers even suggested that learning in the brain is accomplished through changing the synaptic connection strength alone [41, 145–147]. Neuron-type plasticity is currently an experimental idea, where we find papers suggesting possible alteration through external effects like viruses [148], optical stimulations [149], and chemicals [150]. In simulation studies of neuron-neuron interactions, plasticity in neurons types have not been thoroughly explored. It appears therefore that the neuroscience community is starting to explore this phenomenon more seriously, in particular to elucidate the mechanism(s) behind the change in excitability of neurons [151, 152]. Whatever the nature of non-synaptic neuron plasticity, at the level of Izhikevich’s model it must be mediated by terms where the output of a neuron is directly coupled to the parameters of another neuron, even if the mechanism involve synaptic processes. As we can see, the phase diagram in Fig. 3.4 suggests an alternative adaptation process that can have direct implications on information processing if neuron types can change. Therefore, in addition to pulse coupling, we also introduce parametric coupling, where the parameters of one neuron are affected by the outputs of the neurons it is connected to.

In principle, the spiking activities of the neighbors of a neuron change the local electrostatic and chemical environments it is in. This dynamic environment can

push the neuron from one spiking type to another, provided its parameters are close to the phase boundary between the two spiking types. Unfortunately, a simulation at this level of detail would be prohibitively expensive and complicated. Therefore, as a test of concept we adopt a design approach based on an artificial energy landscape. In this framework, we treat the neuron parameter Q as a particle moving in a *phase-diagram-inspired energy landscape*, E , so that it will experience a force F_e due to the potential gradient driving it to the nearest local minimum. The particle also experiences a friction force F_r when moving within this artificial energy landscape. Finally, the particles are connected to other particles through springs, so that they will exert spring forces F_s on each other. With the aid of Verlet integration, we can write these equations of motion mathematically as

$$Q_{n+1} = 2Q_n - Q_{n-1} + Ah^2, \quad (3.2)$$

$$A = A(Q_n) = (F_e + F_s + F_r)/m, \quad (3.3)$$

$$F_s = -K_s \sum_{\substack{\text{neighbours} \\ i \neq j}} (Q^{(i)} - Q^{(j)}) \theta(v^{(j)}), \quad (3.4)$$

$$F_r = -K_r(Q_n - Q_{n-1})/h, \quad (3.5)$$

$$F_e = \frac{dE(Q_n)}{dQ_n}. \quad (3.6)$$

In the above equations, Q parametrizes the straight line going from $(c, d) = (-55, 4)$ to $(c, d) = (-50, 2)$. This effectively reduced the parameters values in Fig. 3.3 to $Q = -2$ for RS, $Q = 0$ for IB, and $Q = 1$ for CH. Here, Q_n are the values of Q at time step $t = n \cdot h$, while $h = 10^{-3}$ is the integration step size, $K_s = 25$ is the spring constant, and $K_r = 1$ is the friction coefficient. The choice of the spring and friction coefficients are such that the parametric interactions are preserved at a reasonable scale and the results do not vary drastically so long the parameters are kept at similar magnitudes. The superscripts in Eq. (3.4) refers to the index of the neuron node. The Heaviside step function $\theta(v^{(j)})$ term in Eq. (3.4) ensures the update of $Q^{(i)}$ occurs only when its neighbour neuron j is spiking.

3.3.1 Phase-Diagram-Inspired Energy Landscape

To construct an artificial energy landscape with a ‘thermodynamic’ equilibrium that is consistent with the phase diagram, we determine the sharp regime boundaries from the various spiking types displayed in Fig. 3.3 and the counting of number of peaks per unit time in Fig. 3.5(*top*), and assign these as energy value of zero in the artificial energy landscape. We then assign a negative energy value whose magnitude is proportional to the width of the spiking regime at the middle of each spiking regime. Once these two sets of points on the artificial energy landscape have been identified, we fit a cubic spline through them to obtain the energy landscape as shown in Fig. 3.5(*bottom*).

While one may realized that it is also possible to simulate parametric dynamics over the full (c, d) parameter space. This would be a lot more involved numerically compared to what we have done in this Chapter. More importantly, an energy landscape over the full (c, d) parameter space would consists of multiple elongated flat directions. A neuron anywhere along a flat direction would be in the same $\text{CH}n$ spiking phase, so the most important aspect of its parametric dynamics would be that normal to the phase boundary. Fortunately, when we chose to restrict our simulations to the straight line parametrized by Q , this straight line is very close to being normal to the set of $\text{CH}n$ phase boundaries. Ultimately, if we want to be strictly normal to all phase boundaries, we would need to restrict our simulations to be along a curve, which would again be more troublesome.

FIGURE 3.5: *Top:* Boundaries of each of the various spiking regimes found by collecting long times series of each of the parameter values from $Q = 0$ to $Q = 2$ (showing only $Q = 0$ to $Q = 1.4$ because beyond this value of Q , the neuron is in the FS regime) in steps of 4×10^{-5} and using peak analysing technique to segregate and count the various peak type observed. The vertical axis represents the number of peaks per unit time, and spiking types are labelled in Roman numbering as illustrated in Fig. 3.3. *Bottom:* Phase Diagram Inspired Energy Landscape obtained by performing a cubic spline of critical turning points of the boundaries and their midpoints. The absolute value of the energies is inconsequential to the dynamics, thus the energy at the boundaries is arbitrarily set to 0 while that of their midpoints is set to the difference of the parameter values at the boundary.

3.3.2 Quasi-Equilibrium Distribution of Parameters

To perform a computation, we need to first be able to represent different information. In a digital computer, this is done by using strings of ‘1’s and ‘0’s, representing the on and off states of logic gates. In principle, other forms of representation can also work. For example, the distinct spiking regimes in the Izhikevich neuron model can be used as a representation. To use such a basis, the spike type must be able to switch from one to another during the course of a computation. Indeed, when the parameters of our coupled neuron system were allowed to vary in the energy landscape, we achieved a quasi-equilibrium distribution of the parameters, as shown in Fig. 3.6.

In this quasi-equilibrium distribution, we can estimate the analogous of a temperature of the system using the ratio of probabilities of parameters in each energy well (known as the Boltzmann distribution) and the energy differences of the wells in the Boltzmann factor: $\frac{P_1}{P_2} \propto \exp\left(-\frac{\Delta E}{T}\right)$. Starting with a ring network of neurons, we find that with increasing rewiring probability, the temperature of the system increases, as seen in Fig. 3.6(c). This means that the system can easily switch from one spike type to another spike type. However, an infinite temperature is not be desirable here, as that would quickly randomize the information.

3.4 Complexity and Small-World Network

Like many other brain scientists, we believe that information encoding in the brain is not done at the level of single neurons, but over a network of neurons [86, 129]. Hence, the final missing ingredient to understanding information encoding in the brain is the topology of the neuron network. For this final part of the study, we chose to work with small-world networks. This network topology allows a neuron to trigger another neuron far away and is even capable of self-sustaining activity [94, 153–155]. Moreover, many papers have also found that the small-world property is a prominent common ingredient in the functional network of the

FIGURE 3.6: (a) Frequency distribution of parameters with the gray dotted line marking the transition boundaries of parameters seen in Fig. (3.5). This data is collected over 40 sets of network of size, $N = 200$, and number of nearest neighbour, $k = 3$. The inset on the top right is the zoom-in showing the spread of the parameters about the peaks of their distribution. (b) Ratio of probabilities of parameters in each energy well against the energy differences of the wells. (c) Segregated by different network rewiring probabilities, p .

human brain [91–93]. In particular, Sporns *et al.* found by generating a family of networks that the complexity of the static network topology is maximized in small-world-like networks, and even more so in network topologies derived from actual neuroanatomical data [156].

To set up our small-world network, we used the Watts-Strogatz [157] *rewiring algorithm* to generate networks with $N = 200$ nodes, each connected to $2k$ other nodes. This algorithm allows us to continuously tune the network from a regular ring network (with N nodes each connected to the nearest k neighbours) to a fully random network, by adjusting the probability, p , of rewiring the connections

(see Fig. 3.7). Newman and Watts also proposed an alternative way to link distant nodes, by starting from the regular network, and adding shortcuts between randomly chosen pairs of nodes [158]. We call this the *shortcut algorithm*.

FIGURE 3.7: Effects of rewiring probability. Three networks with $N = 200$ nodes each connected to the nearest $k = 3$ neighbours with increasing rewiring probabilities, (a) $p = 0$ (regular ring), (b) $p = 1/32$ (small-world), and (c) $p = 1/2$ (random).

3.4.1 Complexity Measures

The hypothesis we would like to test in this section is that the brain is in a self-organized-critical state, within which it has a large complexity, and also a small-world network topology. Self-organised criticality (SOC), proposed by Per Bak in 1987 [159, 160], is a concept closely intertwined with complexity [161]. However, while the concept of SOC is well established and clearly defined [159, 160, 162], that of complexity is generally understood but lacks unanimous definition [163]. Many have offered methods to compute or estimate the complexities of dynamical systems [138, 164]. In particular, using the ϵ -machine representation of a process, Crutchfield showed that complexity is maximised in dynamical systems as they approach the edge of chaos as well as other systems at criticality [18, 138].

With this in mind, we seek to find the parallel phenomenon of complexity peaking between order and disorder in our computational model of neuron dynamics. However, instead of measuring Crutchfield's ϵ -machine complexity, we measured the statistical complexity C_{LMC} introduced by Lopez-Ruiz, Mancini, and Calbet [164]. This C_{LMC} complexity measure is a good estimate of complexity based on a

probabilistic description of the dynamics, and is also computationally lightweight in comparison to Clutchfield's ϵ -machine. C_{LMC} is also rigorously studied [163, 165, 166] and choice of candidate for further improvisation [167] and generalisation [168].

For a system with N accessible states $x_1, x_2, \dots, x_N$ each with corresponding probabilities $p_1, p_2, \dots, p_N$, the C_{LMC} complexity is defined as

$$C_{LMC} = H \cdot D, \quad (3.7)$$

where

$$H = - \sum_{i=1}^N p_i \log p_i \quad (3.8)$$

is the Shannon entropy, a measure of disorderedness, and

$$D = \sum_{i=1}^N (p_i - 1/N)^2 \quad (3.9)$$

is named as the *disequilibrium*. With this definition we have the complexity $C_{LMC} = 0$ in both a fully ordered system where the entropy $H = 0$, and in a equilibrium system where the probabilities of the states are uniformly random $D = 0$ despite having high entropy H . For any other system the complexity C_{LMC} will have a value higher than zero which represents that the system is neither completely ordered nor in equilibrium (see Fig. 3.8 *Left* for more details on the calculations of C_{LMC}). The relationship between C_{LMC} and H gets complicated quickly with larger systems ($N > 2$). Fig. 3.8 *Right* illustrate a $N = 3$ states system where multiple combinations of probabilities p_i results in the same magnitude of entropy H yet different complexities C_{LMC} . However the fact remains that C_{LMC} does not grow indefinitely with entropy H .

FIGURE 3.8: Illustration on Statistical Complexity. *Left*: Plot of the statistical complexity C_{LMC} , entropy H and disequilibrium D versus the probability of obtaining a head outcome in a biased coin ($N = 2$ states) system. To calculating C_{LMC} in a 2-state system we can simply rewrite Eq. (3.8) and Eq. (3.9) as $H = -[p(\text{head}) \log p(\text{head}) + p(\text{tail}) \log p(\text{tail})]$ and $D = \{[p(\text{head}) - 1/2]^2 + [p(\text{tail}) - 1/2]^2\}$. Since we have $p(\text{head}) = 1 - p(\text{tail})$, the graph is symmetrical about $p(\text{head}) = 0.5$. *Right*: Plot of the statistical complexity C_{LMC} versus the normalised entropy $\bar{H}$ for $N = 2$ and $N = 3$ states systems.

3.4.2 Maximising of Complexity

The key network parameters to define a small-world network is the local clustering coefficient C measuring the cliquishness of a typical neighbourhood and characteristic path length L measuring the typical separation between two vertices in the graph [157]. More concretely, C_i of a particular i^{th} -node is calculated by the proportion of links between the vertices within its neighbourhood divided by the number of links that could possibly exist between them, hence the network average clustering coefficient is $C(p) = \frac{1}{N} \sum_{i=1}^N C_i$ where p is the rewiring probability tuning the network topology and N is the total number of nodes. As for the characteristic path length we have $L(p) = \frac{1}{N(N-1)} \sum_{i \neq j} d(i, j)$ where $d(i, j)$ is the shortest distance between i^{th} and j^{th} node.

As we tune the $N = 200$ and $k = 3$ network by increasing the rewiring probability p , we find in Fig. 3.9 that the characteristic path length $L(p)$ decreases rapidly, whereas the average clustering coefficient $C(p)$ remains large and only starts its rapid decrease after p becomes large enough. The network is close to being regular when $L(p)$ and $C(p)$ are both large, and is close to being random when $L(p)$

and $C(p)$ are both small. In between these two extremes, the network has small $L(p)$, but large $C(p)$, and is manifestly small-world. Therefore, the ratio $S_w(p) = C(p)/L(p)$ has a peak at intermediate rewiring probability p . When we apply the shortcut algorithm, $L(p)$ has the same behavior as the probability p of adding a shortcut increases. However, $C(p)$ remains large even at large p , so there is no well-defined peak in $S_w(p)$.

FIGURE 3.9: *Left axis:* Plot of clustering coefficient $C(p)/C(0)$ and characteristic path length $L(p)/L(0)$ of the networks, both scaled so that their maxima are one, with varying rewiring probabilities, p . The ratio between these two properties, $S_w(p) = C(p)/L(p)$, is indicative of the transition to a small world network. In this figure, $S_w(p)$ is also scaled so that its maximum is one. *Right axis:* Plot of the statistical complexity C_{LMC} versus the rewiring probability p , averaged over the parameter time series $Q^{(i)}(t)$ of the $i = 1, \dots, 200$ neurons. We do not show the clustering coefficient $C(p)$ versus varying probability p to add a shortcut because the original k -nearest-neighbor network was not destroyed by adding shortcuts, even at very high p , and thus the resulting network is not fully random. *Note:* Connecting lines were added the data points only to enhance visibility of the trends.

The calculation of the C_{LMC} statistical complexity is obtained first by reducing the dynamics of $Q^{(i)}(t)$ to its symbolic dynamics of the nine CH regimes (indicated

by Roman numerals in Fig. 3.3. This is done by computing the averages of $Q^{(i)}(t)$ over non-overlapping time windows of 50 time steps each. Since one time step is equivalent to 1×10^{-3} time unit, the size of each time window is equivalent to $\Delta t = 0.05$. Secondly, a scale of 4 is chosen to analyse as states of the system resulting in a total of 9^4 possible states, which is comparable to using a scale of 12 for $x < \frac{1}{2}$ and $x > \frac{1}{2}$ in the logistic map, i.e. 2^{12} states [164]. Lastly, we collect the symbolic time series of all $N = 200$ neurons over the 10,000 time windows, and over 40 different initial conditions, to determine the probabilities of each of the 9^4 symbolic state. With this probability distribution, we calculate the values of H , D and C using Eq. (3.7) to Eq. (3.9).

Moving from an ordered ring network to a random network, we demonstrated in Fig. 3.9 that the statistical complexity $C_{LMC}(p)$ of a network of parameter-coupled neurons evolving within a phase diagram inspired energy landscape peaks close to the peak of $S_w(p)$. In other words, the peak in statistical complexity coincides with the peak in small-world character of the neuron network. This result suggests that for the brain to maximize its statistical complexity at criticality for effective information processing, it should also have the strongest small-world character.

3.5 Chapter Conclusions

In this Chapter, we explored the plausibility of information processing in brains that are at criticality, and how this information processing is related to the empirically observed small-world topology of brain functional networks. We do this by selecting the CH region in the phase diagram of a Izhikevich neuron. This has the characteristics of being at the edge of chaos, i.e. many distinct CH regimes are accessible within a small parameter region. We then couple Izhikevich neurons such that the output of a neuron changes the parameters of neurons it is coupled to. We called this *parametric coupling*, and simulated the dynamics of the set of neuron parameters on a phase diagram inspired energy landscape.

Using the rewiring algorithm of Watts and Strogatz, we interpolate between regular neuron networks, small-world neuron networks, and random neuron networks, and find that the statistical complexity of the neuron dynamics peaks when the small-world character of the network also peaks. This suggests that the small-world character of brains is connected to the statistical complexity of brains sitting at criticality.

With this proof of concept, we now have the foundation to move the study forward to understanding how information can be actually encoded and processed within such neuron networks. We would then move on to identifying possible built-in logic gates in the network that can manipulate information. These should be small but over-represented dynamical motifs that appear in brain information processing. With these, we aim to better understand the computational capability of the brain at criticality. We believe that in science, it is not only important to obtain results, but also to ask the right questions that would frame the problem. What it means for the brain to be criticality? Why is it at criticality? How does it get to criticality? What can it do at criticality? The entire set of questions constitutes an ambitious research program that would take much time to be completed. The results we present in this Chapter partially answers some of these questions, and should be appreciated in this light.

Chapter 4

Robust Clustering Analysis of Synchronisation in Small-world Brain Networks

From Chapter 3, we understood that the neuronal network on a Small-World topology contains much potential for information processing in terms of statistical complexity. To build on this idea further, we would like to understand how the brain stores and processes information, and hopefully identify key mechanisms that demonstrate such capabilities.

Synchronisation in neuronal activation as covered in Chapter 2, is the key indicator of brain functional activity. Here our challenge is finding a good and reasonable measure of synchrony of the spiking times series for our project. Phase analysis method such as the Hilbert transform [105, 106] discussed earlier in Chapter 2 were tested but cannot be use in our study, because it is too expensive computationally to store high-resolution time series generated by our simulations, and the technical difficulty to perform such analysis ‘on-the-fly’ to avoid storing the full time series. Another issue we faced regarding the use of the phase analysis methods is the long ‘resting’ state of the neurons versus the short ‘spiking’ state. This results in poor

detection of the phase-lock when calculating the synchrony as the spike duration is heavily outweighed by the ‘resting’ state.

In this chapter, we demonstrate the viability of using spike times series distance measuring technique to better identify synchronisation activities. We then used hierarchical clustering to identify the synchronisation clusters, and visualized the flow of such synchronization activities in the form of alluvial diagrams.

4.1 Pairwise ISI-distance

Kreuz *et al.* developed the “interspike-intervals (ISI)” distance measure [107, 108] that is independent of rest time as it capture only the moments a neuron spikes. Other sophisticated distance measuring methodologies exist [103, 109], but we tested the ISI distance, and found that it was good enough for us to detect synchrony.

To measure the distance between a pair of neurons, Kreuz *et al.* first denoted the spike times as $t_i^n = t_1^n, \dots, t_{M_n}^n$ where M_n is the number of spikes for neuron n with $n = 1, \dots, N$. They then defined the current interspike interval for any time instant $t_1^n < t < t_{M_n}^n$ to be

$$x_{ISI}^n(t) = \min(t_i^n | t_i^n > t) - \max(t_i^n | t_i^n < t). \quad (4.1)$$

To achieve “a time-resolved, symmetric, and scale-invariant measure of the relative firing-rate pattern” [108], they defined the instantaneous ISI-ratio between any pair of neurons i and j to be

$$I_{ij}(t) = \begin{cases} \frac{x_{ISI}^i(t)}{x_{ISI}^j(t)} - 1 & \text{if } x_{ISI}^i(t) \leq x_{ISI}^j(t) \\ -\left(\frac{x_{ISI}^j(t)}{x_{ISI}^i(t)} - 1\right) & \text{otherwise.} \end{cases} \quad (4.2)$$

This definition allows ISI-ratio to be 0 for identical $x_{ISI}^n(t)$ and approaches -1 or 1 if either of the $x_{ISI}^n(t)$ is vastly different from the other. Since the sign (be

it positive or negative) does not matter for our purpose, the absolute value is considered when the ISI-distance is defined and integrated over time:

$$D_{ij} = \frac{1}{T} \int_{t=0}^T dt |I_{ij}(t)|. \quad (4.3)$$

4.1.1 Sparse Representation of Times Series

The greatest advantage this method affords for our study is that it only requires the spiking times. This allowed us to reduce the time series to binaries with ‘0’ representing the non-spiking state and ‘1’ representing the spiking state. Since spikes are sparse relative to the resting states, we can store the long time series of large networks using the sparse matrix data structure in MATLAB with small and manageable memory size (see Fig. 4.1).

FIGURE 4.1: Sparse Representation of Times Series. A sample of five time series over a period of $t = 1000$ ms were displayed in (a) as full is reduced to the symbolic fire time in (b). The integration step size used here is $h = 0.01$ ms and storing five of such full time series would occupy $8 \times 1000 \div 0.01 = 8$ MB. When we actually simulate the neuronal system, the number of neurons and resolution used will be scaled up and the memory usage will quickly exhaust the computing resource available.

4.2 Robust Hierarchical Clustering

With the pairwise ISI-distances between times series, we can now perform hierarchical clustering to separate and identify groups of neurons participating in synchronous activity. Hierarchical clustering is a widely-used and well-known textbook method [169, 170], and has several algorithms such as the *single linkage* (SLHC) computing the shortest distance between elements of two clusters, the *complete linkage* computing the furthest distance between elements of two clusters, and lastly the *average linkage* (ALHC), computing the average distances between elements of two clusters. Quoting from MATLAB’s documentation on hierarchical clustering [171], these algorithms are used to “link pairs of objects that are in close proximity”, and as individual elements “are paired into binary clusters, the newly formed clusters are grouped into larger clusters until a hierarchical tree is formed”.

FIGURE 4.2: *Left:* A dendrogram for one of the time windows. The convoluted x -axis is the label of each individual element which is this case is our neuron. The y -axis is the linkage distance d_l between the elements their clusters obtained through the complete-link hierarchical clustering (CLHC) algorithm. *Right:* Colormap plot of the pairwise ISI-distance matrix with the elements sorted according to the CLHC algorithm. Similarly here, the convoluted axes labels are the individual elements.

Hierarchical clustering give a fairly good picture of what synchronization clusters are present in a given time window after reordering the neurons in accordance their clusters. However, even after the reordered pairwise ISI-distance matrix is shown as a colormap (see Fig. 4.2), we see relatively strong cross correlations in the off diagonal blocks. This is a well known limitation of the hierarchical clustering algorithm [172–175]. We can be more certain that we are tracking the right groups of neurons when we perform sliding window analysis (see 4.3) by using an additional procedure developed by Teh *et al.* [175] to discover the robust clusters. The idea revolves around the fact that these strong off-diagonal correlations tells us how clusters interact with each other, and these interactions should be taken into account when we search for robust clusters. Teh *et al.* defines

a robust cluster as a cluster whose components remain unchanged over a wide range of depth in the dendrogram and the robust length of a cluster l_{robust} as the linkage difference between the hierarchy level where it is first formed and the hierarchy level where it is merged with another cluster [175]. As shown in Fig. 4.3, this robustness length involving three clusters $\mathbf{cl}_i$, $\mathbf{cl}_j$, and $\mathbf{cl}_k$ can be described mathematically as

$$l_{robust}(\mathbf{cl}_i \cup \mathbf{cl}_j) = d(\mathbf{cl}_i \cup \mathbf{cl}_j, \mathbf{cl}_k) - d(\mathbf{cl}_i, \mathbf{cl}_j), \quad (4.4)$$

where $d(\mathbf{cl}_i, \mathbf{cl}_j)$ is the linkage distance when $\mathbf{cl}_i$ merged with $\mathbf{cl}_j$ to form $\mathbf{cl}_i \cup \mathbf{cl}_j$, and $d(\mathbf{cl}_i \cup \mathbf{cl}_j, \mathbf{cl}_k)$ is the linkage distance when $\mathbf{cl}_i \cup \mathbf{cl}_j$ merged with another cluster $\mathbf{cl}_k$ to form $\mathbf{cl}_i \cup \mathbf{cl}_j \cup \mathbf{cl}_k$.

FIGURE 4.3: An illustration of how the robust length of a cluster l_{robust} is calculated from a dendrogram.

Finally, we define a cluster to be robust if the the following conditions are fulfilled:

$$\begin{aligned}
 \text{I} \quad & l_{robust}(\mathbf{cl}_i \cup \mathbf{cl}_j) > \delta_{Robust}, \\
 \text{II} \quad & d(\mathbf{cl}_i, \mathbf{cl}_j) \leq d_{Max}, \\
 \text{III} \quad & d(\mathbf{cl}_i, \mathbf{cl}_j) \geq d_{Min}.
 \end{aligned} \tag{4.5}$$

These conditions allows an additional control on the acceptable threshold δ_{Robust} for the robust length l_{robust} to be considered robust, and d_{Min} to control and prevent the fragmentation of close clusters while d_{Max} allows automatic fragmentation of clusters far apart.

4.3 Sliding Windows Analysis

Sliding windows analysis has been in use for a long time [176] is sometimes also known as rolling window [177]. The actual implementation in this Chapter is to “chop” the time series into over lapping blocks as illustrated in Fig. 4.4. This approach allows us to control two features namely the window size and the sliding step size.

FIGURE 4.4: Illustration of the sliding windows approach used in this Chapter. The red dotted box depicts the window of the time series cropped for analysis and is slide along the time axis over a constant amount to crop the next time window. In this figure, the window size is $t = 300\text{ms}$ and the sliding step size is $t = 150\text{ms}$.

4.4 Information Flow Visualisation: Alluvial Diagram

By now we have identified the robust synchronised clusters in each time window. The next step is to observe how the clusters change with time, as the neuronal network processes information. However, working the lists of robust synchronised clusters (in Eq. (4.6)) over many time windows does not allow us to easily sieve out the most useful information. Mathematically, a robust synchronised clusters

list of multiple time windows (RBC-list) is written as:

$$\begin{aligned} \text{RBC-list} &= \{\mathbf{RBC}(t_1), \mathbf{RBC}(t_2), \dots, \mathbf{RBC}(t_w)\} \\ \mathbf{RBC}(t_1) &= \{\mathbf{cl}_1^{(t_1)}, \mathbf{cl}_2^{(t_1)}, \dots, \mathbf{cl}_n^{(t_1)}\} \\ \mathbf{cl}_1^{(t_1)} &= \{\epsilon_1, \epsilon_2, \dots, \epsilon_k\} \end{aligned} \quad (4.6)$$

where $t_1, t_2, \dots, t_w$ are labels for the time windows (described in Section 4.3), $\mathbf{RBC}(t_w)$ is the robust synchronised clusters list at t_w and each $\mathbf{cl}_n^{(t_w)}$ are the robust synchronised cluster containing the $\epsilon_1, \epsilon_2, \dots, \epsilon_k$ elements identified earlier in Section 4.2.

TABLE 4.1: Elements of the three clusters at time t_1 and of the two clusters at time t_2 .

$n \backslash t_w$	t_1	t_2
1	$\{1, 2, 3, 4\}$	$\{1, 2, 6, 8, 9\}$
2	$\{5, 6, 7\}$	$\{3, 4, 5, 7\}$
3	$\{8, 9\}$	

For illustration purposes, suppose we have only two time windows, t_1 with three clusters $\mathbf{cl}_1^{(t_1)} = \{1, 2, 3, 4\}$, $\mathbf{cl}_2^{(t_1)} = \{5, 6, 7\}$, $\mathbf{cl}_3^{(t_1)} = \{8, 9\}$ and t_2 with two clusters $\mathbf{cl}_1^{(t_2)} = \{1, 2, 6, 8, 9\}$, $\mathbf{cl}_2^{(t_2)} = \{3, 4, 5, 7\}$ (see Table 4.1). Then, the RBC-list in Eq. (4.6) will become

$$\text{RBC-list} = \left\{ \begin{bmatrix} 1 & 5 & 8 \\ 2 & 6 & 9 \\ 3 & 7 & 0 \\ 4 & 0 & 0 \end{bmatrix}, \begin{bmatrix} 1 & 3 \\ 2 & 4 \\ 6 & 5 \\ 8 & 7 \\ 9 & 0 \end{bmatrix} \right\}. \quad (4.7)$$

Note that clusters are of different sizes (i.e. the number of elements is not the same) therefore when placed in a matrix, the empty entries are padded with zeros. In the above illustration, there are only two time windows with a total of nine elements. In a typical simulation we would use at least 200 neurons (each represented as one element) and at least 10 time windows. It would be impractical to manually read through the list of elements and make a meaning observation.

Fortunately, visualising structural change in large complex networks can be done using a class of diagrams called the *alluvial diagrams*, where blocks represent clusters of nodes and connecting strips between the blocks represent flows of the composition of these clusters over time (see Fig. 4.5). The size of the block represents the size of the cluster and the thickness of the connecting strips represents the number of common components contained in both connected blocks. In particular, Rosvall and Bergstrom highlighted and summarized the significant structural changes in citation patterns with alluvial diagrams [178]. The alluvial diagrams capture important structural changes and can be further emphasised by colouring to ease the identification of major transitions.

FIGURE 4.5: Alluvial diagrams illustrating structural change in a system of clusters over two time windows. Three key structural changes shown here are (a) the splitting of a cluster into smaller clusters, (b) merger of smaller clusters into a bigger cluster, and lastly (c) a mixture of merger and separation which is called exchange in this case. Note that the number written in each cluster here denotes the cluster size (i.e. the number of elements within the cluster). These plots are done in MATLAB and the codes are translated from a Python source [179] and can be found in Appendix 4.7.

To construct an alluvial diagram, we need to extract two information from the RBC-list in Eq. (4.6). Firstly we need to get the size of the clusters by counting the number of elements in each cluster and for each time window. We call this the MS-list. Secondly we extract the relationship matrices $\mathbf{rl}_{t_{w-1} \rightarrow t_w}$ that maps the number of elements from clusters in the previous time window to clusters in the next time window. Each relationship matrix is filled by going through

all the clusters in the next time window, and for each cluster there, count the number of matches in the list of elements in the cluster $\mathbf{cl}_n^{(t_w)}$ to all the clusters in the previous time window $\{\mathbf{cl}_1^{(t_{w-1})}, \mathbf{cl}_2^{(t_{w-1})}, \dots, \mathbf{cl}_n^{(t_{w-1})}\}$. To illustrate continuing from the example in Eq. (4.7) we would have

$$\begin{aligned} \text{MS-list} &= \left\{ \begin{bmatrix} 4 & 3 & 2 \end{bmatrix}, \begin{bmatrix} 5 & 4 \end{bmatrix} \right\} \\ \mathbf{rl}_{t_1 \rightarrow t_2} &= \left\{ \begin{bmatrix} 2 & 2 \\ 1 & 2 \\ 2 & 0 \end{bmatrix} \right\}. \end{aligned} \quad (4.8)$$

With the MS-list and all the $\mathbf{rl}_{t_{w-1} \rightarrow t_w}$, we then plot the boxes with sizes corresponding to the MS-list, and use cubic Bézier curves of the form

$$\mathbf{B}(t) = (1-t)^3 \mathbf{P}_0 + 3(1-t)^2 t \mathbf{P}_1 + 3(1-t) t^2 \mathbf{P}_2 + t^3 \mathbf{P}_3, 0 \leq t \leq 1 \quad (4.9)$$

to draw the connecting ribbons corresponding to $\mathbf{rl}_{t_{w-1} \rightarrow t_w}$ shown in Fig. 4.6. $\mathbf{P}_0$, $\mathbf{P}_1$, $\mathbf{P}_2$, and $\mathbf{P}_3$ are the four control points with $\mathbf{P}_0$ and $\mathbf{P}_3$ being the respective starting and ending points, and $\mathbf{P}_1$, $\mathbf{P}_2$ being the directional guide such that the distance between $\mathbf{P}_1$ and $\mathbf{P}_2$ determines how close the curve approaches $\mathbf{P}_1$ before turning to $\mathbf{P}_2$. The actual mathematical implementation of this Eq. (4.9) is not critical to this Chapter but used to generate smooth graphical curve and is included here for completeness. We have implemented this plot function in MATLAB and the codes are translated from a Python source [179] and can be found in Appendix 4.7.

FIGURE 4.6: Example from Eq. (4.7) and Eq. (4.11) as an alluvial diagram. The texts included in the blocks of figure are descriptive labels of the members of the elements within each cluster and its size. The texts included in the beginning of each connecting ribbon are descriptive labels of the elements moving to the cluster in the next time window.

As seen in Fig. 4.6, the alluvial diagram can also include the information on the elements contained within the clusters and allows references to the participating neurons when needed. This way, tracing and identification of key prominent neurons participating in turning points of events or behaviours like logical gates is made possible.

4.4.1 Tracing a Stream of Robust Clusters

FIGURE 4.7: Alluvial diagram for a Izhikevich Neuron Network. This diagram is constructed based on the simulation results of a $N = 200$ neurons network with $K = 3$ nearest neighbour connections rewired at $p = 0.1$ over a total of $t = 770$ ms.

A raw plot of the alluvial diagram for our actual simulation results yielded a mess (see Fig. 4.7). Because of the sheer number of clusters and connecting ribbons, it is hard to identify key features in the cluster flows. To improve the visibility of the robust clusters and the exchanges of nodes between them sorting had to be done. We would like to keep the positions of the prominent robust clusters relatively unchanged across the time windows such that the major connecting paths remains visibly *thick*. When two clusters of adjacent time windows are far apart in the vertical axis, the connecting line appears *thin* regardless of size of exchange due to the visual artefact from the alluvial diagram algorithm.

To perform sorting, we first rearrange the positions of the clusters $\mathbf{cl}_n^{(t_w)}$ based on their sizes from the largest to smallest for the last time window. We then back-traced the list of the robust clusters that are the most strongly connected in the previous time windows, i.e. $\max(\mathbf{rl}_{t_{w-1} \rightarrow t_w})$, to each of the clusters at the last time window. We then sort the positions of the clusters $\mathbf{cl}_n^{(t_{w-1})}$ in the time window before the last time window giving descending priority with respect to the size of the connecting ribbon $\mathbf{rl}_{t_{w-1} \rightarrow t_w}$. Lastly, we traced the the robust clusters path, once again based on the largest inter-connecting neurons $\max(\mathbf{rl}_{t_{w-1} \rightarrow t_w})$, but this time allowing up one split path if it shares the same size as the first largest path. This path is coloured in red and the first of such path is shown in Fig. 4.8 and the rest can be found in Appendix 4.8.

FIGURE 4.8: Sorted alluvial diagram for a Izhikevich Neuron Network. This diagram is sorted from Fig. 4.7 and the red coloured clusters are the first of the many traced robust clusters path based on the largest inter-connecting neurons. See Appendix 4.8 for more tracing of the prominent paths.

In our simulation of the brain network, we do see some strong clusters of neurons activity but not the converging or diverging phenomena that indicate probable logical operations. We believe this is because the system of neurons here are supplied with a constant ambient input potential that mimics a spontaneous resting state activity of a brain. Just like with a resting state fMRI data, we can map the functional architecture of the brain using methods covered in Chapter 2, we can do the same here. However, since we have a simulation model, we should be able to do better. In fact, we can introduce input signals to our neuron network, follow how information from the input signal propagate through the network, to identify logical operations that are possible in such neuron networks.

4.4.2 Input Signal

In this subsection, we describe how we can go beyond resting-state brain networks and work with input signals to the networks. Like a control in an experiment, we can compare the difference between the setup with input signal against the previous ambient rest state to isolate the effects of synchronization in the neural networks and how information propagates through them. In actual brain this would be a task-based study where the subject is assigned to perform certain tasks.

We would like to perform binary control of a certain number of neurons so that we have the ‘on’ state with input current, $I = 7$ or the ‘off’ state with $I = 4$. These values are use in accordance the initial study discussed early in Chapter 2 and the preliminary work in Chapter 3. The neurons can either be selected at random, or they can be groups of neurons in close proximity. With these ‘on’ and ‘off’ states we can examine three cases: ① an ‘on’ network is turned ‘off’; ② an ‘off’ network is turned ‘on’; and ③ the network is periodically turned ‘on’ and ‘off’. Even with these three simple cases, we need to run a large number of simulations to probe how different synchronization pathways are activated in our $N = 200$ neuron network. With the number of simulations we have run thus far, we are unfortunately unable to draw broad, statistically significant conclusions on

all three cases. Here we present preliminary results where we send input signals through two randomly selected control neurons. A control neuron is one in which $I(t)$ is set to the input signal, and this cannot be changed by feedback to the neuron. The results shown in Fig. 4.9 do not seem to show the propagation of the signal. We believe this might be due to the small number of control neurons, or the low frequency of the alternating ‘on’ and ‘off’ signal. In principle we would like to investigate the effect with larger number of control neurons and with groups of close proximity.

FIGURE 4.9: Sorted Alluvial Diagram for a Controlled Izhikevich Neuron Network. This diagram is constructed based on the simulation results of a $N = 200$ neurons network with $K = 3$ nearest neighbour connections rewired at $p = 0.075$ over a total of $t = 1900$ ms with 2 randomly selected neurons as control. The control signal used here is the alternating ‘on’ and ‘off’ at periods of 100 ms. The blocks coloured in blue are the clusters containing the 2 controlled neurons and when the 2 controlled neurons are in the same cluster, the block is coloured red. The green bars at the bottom are indicative of the averaged input signal levels over that particular time window. There are four levels with highest bar representing the time window when controlled neurons are ‘on’ the entire time (100%) and the subsequent levels represents 50%, 25%, and 0%.

4.5 A Quantitative Approach to Validate: Adjusted Rand Index

Apart from sieving through the various alluvial diagrams manually, we can filter out systems of interest by having a scoring system to capture how frequently the neurons are exchanged in the clusters identified. Here, we used a popular metric from the machine learning community, the adjusted Rand index.

The Rand Index computes a similarity measure between two clusterings originally developed to compare various results of different clusterings on the same data set [180]. Here, we used the Rand index to compare the similarity of the clusterings across different time windows. Using the same notation from earlier this Chapter, the Rand index, RI , is

$$RI(\mathbf{RBC}(t_1), \mathbf{RBC}(t_2)) = \frac{a + b}{a + b + c + d} = \frac{a + b}{\binom{N}{2}} \quad (4.10)$$

where

- a is the number of pairs of elements remaining in the same cluster in t_1 and t_2 ,
- b is the number of pairs of elements remaining in different clusters in t_1 and t_2 ,
- c is the number of pairs of elements that were in the same cluster in t_1 but in different clusters in t_2 , and
- d is the number of pairs of elements that were in different clusters in t_1 but in the same cluster in t_2 .

Since $a + b + c + d$ is the total number of pairs, it can be replaced by $\binom{N}{2}$ where N is the total number of elements. Applying this measure to the example from Table 4.1 we have $a = 4$ pairs of elements that remain in the same cluster in t_1 and t_2 :

$$(1, 2) \quad (3, 4) \quad (5, 7) \quad (8, 9)$$

and we have $b = 14$ pairs of elements that remain in different clusters in t_1 and t_2 :

$$\begin{array}{cccccc} (1, 5) & (2, 5) & (3, 6) & (4, 6) & (5, 8) & (7, 8) \\ (1, 7) & (2, 7) & (3, 8) & (4, 8) & (5, 9) & (7, 9) \\ & & (3, 9) & (4, 9) & & \end{array}$$

Therefore, with a total number of $\binom{9}{2} = 36$ pairs, we have $RI = \frac{4+14}{36} = 0.5$.

The adjusted Rand index ARI [181] improves the contrast of the index by discounting the expectation value of the Rand index $\mathbb{E}(RI)$ to have a value close to 0.0 for random labelling of clusters. Mathematically, we have:

$$ARI = \frac{RI - \mathbb{E}(RI)}{\max(RI) - \mathbb{E}(RI)}, \quad (4.11)$$

and this can be evaluated using the contingency table:

$\mathbf{RBC}(t_1) \backslash \mathbf{RBC}(t_2)$	$\mathbf{cl}_1^{(t_2)}$	$\mathbf{cl}_2^{(t_2)}$	$\dots$	$\mathbf{cl}_n^{(t_2)}$	Sums
$\mathbf{cl}_1^{(t_1)}$	s_{11}	s_{12}	$\dots$	s_{1n}	R_1
$\mathbf{cl}_2^{(t_1)}$	s_{21}	s_{22}	$\dots$	s_{2n}	R_2
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
$\mathbf{cl}_m^{(t_1)}$	s_{m1}	s_{m2}	$\dots$	s_{mn}	R_m
Sums	C_1	C_2	$\dots$	C_n	

(4.12)

which summarises the overlap between $\mathbf{RBC}(t_1)$ and $\mathbf{RBC}(t_2)$, and where the entries s_{mn} are the number of same elements in $\mathbf{cl}_m^{(t_1)}$ and $\mathbf{cl}_n^{(t_2)}$. R_m and C_n are the sums of the respective rows and columns of the table. With this, the adjusted Rand index can be expressed as the following:

$$ARI(\mathbf{RBC}(t_1), \mathbf{RBC}(t_2)) = \frac{\sum_{nm} \binom{s_{nm}}{2} - \frac{\sum_m \binom{R_m}{2} \sum_n \binom{C_n}{2}}{\binom{N}{2}}}{\frac{1}{2} [\sum_m \binom{R_m}{2} + \sum_n \binom{C_n}{2}] - \frac{\sum_m \binom{R_m}{2} \sum_n \binom{C_n}{2}}{\binom{N}{2}}}. \quad (4.13)$$

Applying the the adjusted Rand index to the example from Table 4.1, we will get the following contingency table:

$\mathbf{RBC}(t_1) \setminus \mathbf{RBC}(t_2)$	$\mathbf{cl}_1^{(t_2)}$	$\mathbf{cl}_2^{(t_2)}$	Sums
$\mathbf{cl}_1^{(t_1)}$	2	2	4
$\mathbf{cl}_2^{(t_1)}$	1	2	3
$\mathbf{cl}_3^{(t_1)}$	2	0	2
Sums	5	4	

(4.14)

and the adjusted Rand index $ARI = \frac{4-160/36}{13-160/36} = -0.0519$ (3 s.f.). As the example we used here is indeed fairly random in nature, the ARI being close to 0 is a good indication.

Now going back to the two sets of data (resting and controlled state of the Izhikevich neuron network) presented in Section 4.4, we can apply the adjusted Rand index to get a measure of how consistent are the various clusters identified over the overlapping time windows. Similar to the observations in Fig. 4.8, we see that in Fig. 4.10 there is a gradual increase in the consistency of the robust clusters of the resting state neuronal network with the increase in ARI in the later time windows. As for the controlled state neuronal system, we are pleasantly surprised that the ARI (Fig. 4.11) was able to pick up the periodical fluctuations that was not very visible in the alluvial diagram (Fig. 4.9). Although the periodicity of fluctuations in the ARI is not exactly consistent with the averaged input signal level, the effect is definitely distinct from that of the resting state.

With the addition of this quantitative measure of how similar the clusters remained over time, we can quickly select the neuronal network with interesting property to study and to search for the key mechanisms underlying information processing in the brain.

FIGURE 4.10: Adjusted Rand Index of the Resting-State Izhikevich Neuron Network. Using method described in 4.5 applied on data from 4.4.1.

FIGURE 4.11: Adjusted Rand Index of Controlled-State Izhikevich Neuron Network. Using method described in 4.5 applied on data from 4.4.2 and the averaged input signal levels plotting similarly in Fig. 4.9.

4.6 Chapter Conclusions

In this Chapter, we have implemented a collection of methods to analyse the propagation and probable formation of information storage. But using time-resolved distance measure of the time series and a robust hierarchical clustering, these set of procedures can be used not only on simulated data but also on actual recording of brain activity. With high enough precision and resolution, the analytics we described in this chapter can be used to identify how information is processed in the brain in the sense of robust, synchronous, and clustered activities. This chapter demonstrated the capability to trace and track the synchronous activity in an artificial setup. We foresee interesting collaborative work in the future with neuroscientists on experimental data obtained from *in vivo* or *in vitro* setup. This form of synchrony analysis is not restricted to the actual locality nor the phase of the neurons, thus enabling the algorithm to capture “action at a distance” or “long-range connections”, both of which are well studied in neurological activities [94].

4.7 MATLAB function for Alluvial Diagram

Note: These codes translated from a Python source [179] and modified for the purpose of this thesis.

```
function ax = plot_alluvial2a_1c(RBC_list, label_flag, Ctrl)
if ~exist('label_flag','var')
    label_flag = true;
end
R_list = cell(numel(RBC_list)-1,1);
R_label = R_list;
for R_id = 1:numel(R_list)

    A = RBC_list{R_id};
    B = RBC_list{R_id+1};
    R_AB = zeros(size(A,2),size(B,2));
    R_AB_label = cell(size(A,2),size(B,2));
    for i = 1:size(A,2)
        Amem = A(A(:,i)~=0,i);
        for j = 1:size(B,2)
            Bmem = B(B(:,j)~=0,j);
            Lia = ismember(Amem,Bmem);
            R_AB(i,j) = sum(Lia);
            R_AB_label{i,j}=Amem(Lia);
        end
    end
    R_label{R_id} = R_AB_label;
    R_list{R_id} = R_AB;
end
MS_list =cell(size(R_list));% module_sizes_list
for i = 1: length(R_list)
    MS_list{i} = sum(R_list{i},2);
end
MS_list = [MS_list; sum(R_list{end},1)'];
Scts = cell(size(MS_list)); % Special_color_targets
for R_id = 1:numel(R_list)
    accepted_targets = [];
    for C_id = 1:numel(Ctrl)
        [~,jj]=ind2sub(size(RBC_list{R_id}),...
            find(RBC_list{R_id}==Ctrl(C_id)));
        accepted_targets = [accepted_targets,jj];
    end
    Scts{R_id} = accepted_targets;
end
figure;
ax = gca;
set(ax,'xlim',[0,2])
set(ax,'ylim',[-.1,1.1])
set(ax,'xtick',[])
set(ax,'ytick',[])
hold(ax,'on')
box on

MC_list = cell(size(MS_list)); % module_colors_list
mono_color = .3*[1,1,1];
Special_color = [0,0,1];
Special_color2 = [1,0,0];
for i = 1: length(MS_list)
    MC_list{i} = repmat(mono_color,size(MS_list{i},1),1);
    if range(Scts{i})==0
        MC_list{i}(Scts{i}(1),:) = Special_color2;
        continue
    end
    for j = 1:numel(Scts{i})
        MC_list{i}(Scts{i}(j),:) = Special_color;
    end
end
assert( length(MS_list) == length(MC_list))
```

```

RS_list = NaN(size(MC_list)); % ribbon_sizes_list
Max_MS = max(cellfun(@max,MS_list)); % max_modules_size
Sum_MS = max(cellfun(@sum,MS_list)); % sum_module_size
VP_btw = 0.01; % percent of vertical_pad_between_modules
VP_ub = 0.00; % vertical_pad_upper_and_below
HPlr = 0.0; % horizontal_pad_left_right
INS = (1-2*VP_ub-VP_btw*(Max_MS-1))/Sum_MS; % individual_node_size
mw = (1-2*HPlr)/(length(R_list) + 1); % module_width
bw = (1-2*HPlr)/length(R_list); % blank_width
mwnsf = 0.1; % module_width_non_shaded_fraction
Mys_list = cell(size(MS_list)); % module_y_starts_list

for i = 1: length(MS_list)
    MSs = MS_list{i}; % Module_Sizes
    MCs = MC_list{i}; % Module_Colors
    RSs = RS_list(i); % Ribbon_Sizes
    MYS = zeros(size(MSs)); % module_y_starts
    y_now = VP_ub; % Current_y
    xs = HPlr + (i-1) * (mw + bw); % rectangle_x_start

    Number_of_Gaps = length(MSs) - 1;
    Gap_size = (1 - sum(MSs)*INS)/Number_of_Gaps;
    for j = 1:length(MSs)
        MS = MSs(j); % module_size
        color = MCs(j,:);
        MYS(j) = y_now;
        MH = INS * MS; % module_height
        rect_pos = [xs,y_now,mw,MH];
        rectangle('Position',rect_pos,...
            'FaceColor',color,'EdgeColor',.85*[1,1,1])
        if label_flag
            elements = RBC_list{i}(:,j);
            my_text = mat2str(elements(elements>0));
            my_text2 = strcat('n=',num2str(MS));
            text(xs+mw/2,y_now+MH*.75,2,...
                my_text,... % the text
                'FontSize',20,...
                'VerticalAlignment','middle',...
                'HorizontalAlignment','center')
            text(xs+mw/2,y_now+MH*.25,2,...
                my_text2,... % the text
                'FontSize',20,...
                'VerticalAlignment','middle',...
                'HorizontalAlignment','center')
        end
        if ~isnan(RSs)
            rect_pos = ...
                [xs + mw*(i-1)*(1-mwnsf), y_now, mw*mwnsf, MH];
            rectangle('Position',rect_pos,...
                'FaceColor',color,'EdgeColor',.85*[1,1,1])
        end
        y_now = y_now + MH + Gap_size;
    end
    Mys_list{i} = MYS;
end
Mye_list = Mys_list; % module_y_ends_list
CP = 0.6; % curvature_parameter
for t = 1:size(R_list,1)
    for i = 1:size(R_list{t},1)
        for j = 1:size(R_list{t},2)
            RS = R_list{t}(i,j); % ribbon_size
            if RS == 0
                continue
            end
            ys1 = Mys_list{t}(i); % y_start_1
            ye1 = ys1 + RS * INS; % y_end_1
            Mys_list{t}(i) = ye1;
            ys2 = Mye_list{t + 1}(j);
            ye2 = ys2 + RS * INS;
            Mye_list{t + 1}(j) = ye2;
            xs1 = HPlr + (t - 1) * (mw + bw) + mw;

```

```

        xs2 = HPlr + (t + 0) * (mw + bw);
        BV1 = [... % bezier_vertices 1
            [xs1; ys1],... % P0
            [xs1 + CP * bw; ys1],... % P1
            [xs2 - CP * bw; ys2],... % P2
            [xs2; ys2],... % P3
        ];
        BV2 = [... % bezier_vertices 2
            [HPlr + (t - 1) * (mw + bw) + mw; ye1],... % P0
            [HPlr + (t - 1) * (mw + bw) + mw + CP * bw; ye1],... % P1
            [HPlr + (t + 0) * (mw + bw) - CP * bw; ye2],... % P2
            [HPlr + (t + 0) * (mw + bw); ye2],... % P3
        ];
        points = {BV1,BV2};
        if and(any(Scts{t}==i),...
            any(Scts{t+1}==j))
            color1 = [0,0,1];
        else
            color1 = .3*[1,1,1];
        end
        plot_ribbon_using_bezier(ax, color1, points)
        if label_flag
            elements = R_label{t}{i,j};
            my_text = mat2str(elements(elements>0));
            my_text = strcat('>>',my_text);
            text(xs1,ys1+(ye1-ys1)/2,2,...
                my_text,... % the text
                'FontSize',15,...
                'VerticalAlignment','middle',...
                'HorizontalAlignment','left')
        end
    end
end
end
end

function plot_ribbon_using_bezier(ax, color1, points)
    color2=.85*[1,1,1];
    lw=1;
    tRange = linspace(0, 1, 50);
    x_list = cell(2,1);
    y_list = cell(2,1);
    p = cell(4,1);
    for point = 1:2
        for pid = 1:4
            p{pid} = points{point}(:,pid);
        end
        allPoints = p{1}*(1 - tRange).^3 +...
            p{2}*(3*((1 - tRange).^2).* tRange) +...
            p{3}*(3*(1 - tRange).*tRange.^2) +...
            p{4}*tRange.^3;
        x_list{point} = allPoints(1,:);
        y_list{point} = allPoints(2,:);
        plot(ax,allPoints(1,:), allPoints(2,:), 'color',color1,'linewidth',lw)
    end
    hf = fill([x_list{1},fliplr(x_list{2})],...
        [y_list{1},fliplr(y_list{2})],color1);
    set(hf,'EdgeColor',color2)
end

```

4.8 Other Sorted Alluvial Diagrams for a Izhikevich Neuron Network

FIGURE 4.12: In this other sorted alluvial diagram for the Izhikevich neuron network, we track the evolution of synchronies leading eventually to the 12th largest synchronized cluster. From this alluvial diagram, it is clear that this 12th largest synchronized cluster was formed from the merger between two predecessor synchronized clusters.

FIGURE 4.13: In this other sorted alluvial diagram for the Izhikevich neuron network, we track the evolution of synchronies leading eventually to the 9th largest synchronized cluster. From this alluvial diagram, it appears similarly that this 9th largest synchronized cluster was formed from the merger between two predecessor synchronized clusters albeit with noises.

FIGURE 4.14: In this other sorted alluvial diagram for the Izhikevich neuron network, we track the evolution of synchronies leading eventually to the 6th largest synchronized cluster. From this alluvial diagram, it appears that this 6th largest synchronized cluster was formed from the splitting of the first large synchronized clusters.

Chapter 5

Conclusion

In Chapter 1, we frame brain research by comparing the achievements of artificial intelligence (AI) with human cognition. Despite rapid advancement in the AI community, particularly the development of artificial neural network (ANN) based on theoretical neuroscience [182], fundamental research on human brains will continue to be important in shedding light on how the brain works at biological scales. We went on to introduce the idea of the brain operating at the edge of chaos as a hypothesis to explain why the brain can perform at such a biological scale and what it means for the brain to be at the edge of chaos.

In Chapter 2, we reviewed the short history of computing and how digital representation, transformation and logical operations were constructed. We also discussed the parallels of long- and short-term memory in digital computing and in literatures of human cognition. We then moved on to cover more technical topics on non-linear dynamics and how they were used to model the neuron's dynamics. Starting from the earliest known integrate-and-fire model to the famed Hodgkin-Huxley model and lastly the Izhikevich's simplified neuron model we covered the three big classes of neuron model. We then reviewed the advancement in neuroimaging leading to the discovery and application of small-world network and other network theory to study the topological aspects of brain functions. Finally, we reviewed the work done on information processing in the brain, which manifests itself in the form of synchronisation of neural activity.

In Chapter 3, we established the foundation necessary to build a model of the brain at the edge of chaos. We first replicated Izhikevich's work on his simplified neuron model, verifying the spiking behaviours and implemented his model numerically using the 4th-order Runge-Kutta method (RK4). We were able to identify distinct CH-types spikes as we vary the parameter in the c - d plane. With this information, we then mapped out the phase diagram for parameters c and d and identified the region of parameter space to use as our edge of chaos. We used parametric coupling to couple the outputs of these neurons directly to the parameters of other neurons, so that the neuron dynamics can drive transitions from one CH spike type to another on an artificial energy landscape. We also measured the distribution of parameters to verify that our system of neurons are indeed capable of allowing the neurons to dwell in each of the identified CH spike type. Lastly we found that the statistical complexity of the parameter time series is maximised when the neuron network is the most small-world-like, as we tune the network from a regular network to a random network using the Watts-Strogatz rewiring algorithm. Our results suggest that the small-world architecture of neuron connections in brains is not accidental, but may be related to the information processing that they do.

In Chapter 4, we demonstrate the ability to trace information flow in a neuronal network. We used a pairwise interspike-interval distance measure developed by Kreuz *et al.* on the sparse representation of the neuron's voltage time series. From the pairwise distances, we then use the robust hierarchical clustering technique developed by Teh *et al.* to group the neurons. These analysis were done using a sliding window to capture the evolution through time. We then visualise the flow of robust clusters in the form of an alluvial diagram and used the adjusted Rand index to give a quantitative picture. We believe the synchrony analysis developed in this chapter can be used on experimental data obtained by neuroscientist and can further extend the work of existing static or aggregated network theory analyses in neuroscience.

FIGURE 5.1: A concept map illustrating the milestones of the research program.

Future work

Development and Evolution

Development is the process where the young brain grows in complexity into an adult brain. We suspect this process involves the brain's network topology changing from an massive and simply connected hub in young baby's brain to a more complex and multiple-hubs network. With the increasing availability of brain imaging data from baby to adults, if our model can demonstrate the same development process by rewiring the topology of the brain network, this would demonstrate programmed development. Further more, if we devised a method to capture the benefit for development and the competing forces between sensitivity and specificity, we would have achieve a dynamically driven development. Sensitivity and specificity in the context where being sensitive to external stimuli allows the brain to detect small environmental changes but too much sensitivity hinders specific operation such as concentrating on a task at hand.

The summary concept map of this thesis can be seen in Fig. 5.1 with the green items fully completed and the yellow items partially done. Finally we foresee the

---

future direction of this thesis in answering questions regarding development and possibly evolution or modulation of the brain.

Bibliography

- [1] P. J. Werbos, Backpropagation through time: what it does and how to do it. *Proceedings of the IEEE* **78**, 1550–1560 (1990).
- [2] P. D. Wasserman, *Advanced methods in neural computing* (John Wiley & Sons, Inc., 1993).
- [3] D. W. Patterson, *Artificial neural networks: theory and applications* (Prentice Hall PTR, 1998).
- [4] G. Zhang, B. E. Patuwo, M. Y. Hu, Forecasting with artificial neural networks:: The state of the art. *International journal of forecasting* **14**, 35–62 (1998).
- [5] D. Silver, D. Hassabis, Alphago: Mastering the ancient game of go with machine learning, <https://research.googleblog.com/2016/01/alphago-mastering-ancient-game-of-go.html>. Accessed: 2017-07-11.
- [6] Google achieves ai ‘breakthrough’ by beating go champion, <http://www.bbc.com/news/technology-35420579>. Accessed: 2017-07-11.
- [7] R. M. Metcalfe, It’s all in your head the latest supercomputer is way faster than the human brain. but guess which is smarter? *Forbes* **179**, 52 (2007).
- [8] J. Hawkins, S. Blakeslee, *On Intelligence* (Macmillan, 2007).
- [9] S. A. Kauffman, S. Johnsen, Coevolution to the edge of chaos: Coupled fitness landscapes, poised states, and coevolutionary avalanches. *Journal of Theoretical Biology* **149**, 467–505 (1991).

- [10] S. A. Kauffman, Antichaos and adaptation. *Scientific American* **265**, 78–84 (1991).
- [11] S. A. Kauffman, *The Origins of Order: Self-organization and Selection in Evolution* (Oxford university press, 1993).
- [12] S. Kauffman, *At home in the universe: The search for the laws of self-organization and complexity* (Oxford University Press, 1995).
- [13] N. H. Packard, Adaptation toward the edge of chaos. *Dynamic patterns in complex systems* **212**, 293–301 (1988).
- [14] C. G. Langton, Computation at the edge of chaos: Phase transitions and emergent computation. *Physica D: Nonlinear Phenomena* **42**, 12–37 (1990).
- [15] R. Lewin, *Complexity: Life at the edge of chaos* (University of Chicago Press, 1999).
- [16] M. Mitchell, J. P. Crutchfield, P. T. Hraber, Dynamics, computation, and the” edge of chaos”: A re-examination. *arXiv preprint adap-org/9306003* (1993).
- [17] G. Cowan, D. Pines, D. E. Meltzer, *Complexity: Metaphors, models, and reality* (1994).
- [18] J. P. Crutchfield, Between order and chaos. *Nature Physics* **8**, 17–24 (2012).
- [19] K. Kanders, T. Lorimer, R. Stoop, Avalanche and edge-of-chaos criticality do not necessarily co-occur in neural networks. *Chaos: An Interdisciplinary Journal of Nonlinear Science* **27**, 047408 (2017).
- [20] U. Costa, M. Lyra, A. Plastino, C. Tsallis, Power-law sensitivity to initial conditions within a logisticlike family of maps: Fractality and nonextensivity. *Physical Review E* **56**, 245 (1997).
- [21] C. Tsallis, A. Plastino, W.-M. Zheng, Power-law sensitivity to initial conditionsnew entropic representation. *Chaos, Solitons & Fractals* **8**, 885–891 (1997).

- [22] U. Tirnakli, C. Beck, C. Tsallis, Central limit behavior of deterministic dynamical systems. *Physical Review E* **75**, 040106 (2007).
- [23] W. L. Shew, H. Yang, S. Yu, R. Roy, D. Plenz, Information capacity and transmission are maximized in balanced cortical networks with neuronal avalanches. *Journal of neuroscience* **31**, 55–63 (2011).
- [24] J. M. Carlson, J. Doyle, Highly optimized tolerance: A mechanism for power laws in designed systems. *Physical Review E* **60**, 1412 (1999).
- [25] G. O'Regan, *A brief history of computing* (Springer, 2008).
- [26] G. P. Zachary, B. J. Frey, *Vannevar Bush: Engineer of the American Century*, vol. 64 (Free Press, 1997).
- [27] W. Shockley, Transistor amplifier (1954). US Patent 2,666,818.
- [28] S. Humble, Von neumann and computers. *THE MONTANA MATHEMATICS ENTHUSIAST Volume 2, no. 2 [September 2005]* **2**, 118–124.
- [29] A. D. Baddeley, E. K. Warrington, Amnesia and the distinction between long-and short-term memory. *Journal of verbal learning and verbal behavior* **9**, 176–189 (1970).
- [30] A. D. Baddeley, G. Hitch, Working memory. *Psychology of learning and motivation* **8**, 47–89 (1974).
- [31] A. D. Baddeley, N. Thomson, M. Buchanan, Word length and the structure of short-term memory. *Journal of verbal learning and verbal behavior* **14**, 575–589 (1975).
- [32] M. Glanzer, A. R. Cunitz, Two storage mechanisms in free recall. *Journal of Verbal Learning and Verbal Behavior* **5**, 351–360 (1966).
- [33] M. Glanzer, Short-term storage and long-term storage in recall. *Journal of psychiatric research* **8**, 423–438 (1971).
- [34] H. M. Duvernoy, E. A. Cabanis, P. Bourgouin, *The human brain: surface, three-dimensional sectional anatomy and MRI* (Springer Verlag Wien, 1991).

- [35] R. Peyron, B. Laurent, L. Garcia-Larrea, Functional imaging of brain responses to pain. a review and meta-analysis (2000). *Neurophysiologie Clinique/Clinical Neurophysiology* **30**, 263–288 (2000).
- [36] A. E. Cavanna, M. R. Trimble, The precuneus: a review of its functional anatomy and behavioural correlates. *Brain* **129**, 564–583 (2006).
- [37] S. Zola-Morgan, L. Squire, Neuroanatomy of memory. *Annual review of neuroscience* **16**, 547–563 (1993).
- [38] B. Raymond, Short-term storage and long-term storage in free recall. *Journal of Verbal Learning and Verbal Behavior* **8**, 567–574 (1969).
- [39] D. A. Cohen, A. Pascual-Leone, D. Z. Press, E. M. Robertson, Off-line learning of motor skill memory: a double dissociation of goal and movement. *Proceedings of the National Academy of Sciences of the United States of America* **102**, 18237–18241 (2005).
- [40] K. Ericcson, W. G. Chase, S. Faloon, Acquisition of a memory skill. *Science* **208**, 1181–1182 (1980).
- [41] M.-S. Rioult-Pedotti, D. Friedman, G. Hess, J. P. Donoghue, Strengthening of horizontal cortical connections following skill learning. *Nature neuroscience* **1**, 230–234 (1998).
- [42] M. Barinaga, New clues to how neurons strengthen their connections. *Science* **284**, 1755–1757 (1999).
- [43] L. M. Pecora, T. L. Carroll, Synchronization in chaotic systems. *Physical review letters* **64**, 821 (1990).
- [44] G. Chen, X. Dong, From chaos to order perspectives and methodologies in controlling chaotic nonlinear dynamical systems. *International Journal of Bifurcation and Chaos* **3**, 1363–1409 (1993).
- [45] R. E. Mirollo, S. H. Strogatz, Synchronization of pulse-coupled biological oscillators. *SIAM Journal on Applied Mathematics* **50**, 1645–1662 (1990).

- [46] S. H. Strogatz, From kuramoto to crawford: exploring the onset of synchronization in populations of coupled oscillators. *Physica D: Nonlinear Phenomena* **143**, 1–20 (2000).
- [47] J. Jeong, Eeg dynamics in patients with alzheimer’s disease. *Clinical neurophysiology* **115**, 1490–1505 (2004).
- [48] C. J. Stam, Nonlinear dynamical analysis of eeg and meg: review of an emerging field. *Clinical Neurophysiology* **116**, 2266–2301 (2005).
- [49] C. J. Stam, J. C. Reijneveld, Graph theoretical analysis of complex networks in the brain. *Nonlinear biomedical physics* **1**, 3 (2007).
- [50] T. Shajahan, S. Sinha, R. Pandit, Spiral-wave dynamics depend sensitively on inhomogeneities in mathematical models of ventricular tissue. *Physical Review E* **75**, 011929 (2007).
- [51] L. F. Abbott, Lapiques introduction of the integrate-and-fire model neuron (1907). *Brain research bulletin* **50**, 303–304 (1999).
- [52] N. Brunel, M. C. Van Rossum, Lapiques 1907 paper: from frogs to integrate-and-fire. *Biological cybernetics* **97**, 337–339 (2007).
- [53] N. Brunel, V. Hakim, Fast global oscillations in networks of integrate-and-fire neurons with low firing rates. *Neural computation* **11**, 1621–1671 (1999).
- [54] A. N. Burkitt, A review of the integrate-and-fire neuron model: I. homogeneous synaptic input. *Biological cybernetics* **95**, 1–19 (2006).
- [55] A. N. Burkitt, A review of the integrate-and-fire neuron model: II. inhomogeneous synaptic input and network properties. *Biological cybernetics* **95**, 97–112 (2006).
- [56] R. Brette, W. Gerstner, Adaptive exponential integrate-and-fire model as an effective description of neuronal activity. *Journal of neurophysiology* **94**, 3637–3642 (2005).

- [57] A. L. Hodgkin, A. F. Huxley, A quantitative description of membrane current and its application to conduction and excitation in nerve. *The Journal of physiology* **117**, 500 (1952).
- [58] R. FitzHugh, Impulses and physiological states in theoretical models of nerve membrane. *Biophysical journal* **1**, 445–466 (1961).
- [59] C. Morris, H. Lecar, Voltage oscillations in the barnacle giant muscle fiber. *Biophysical journal* **35**, 193–213 (1981).
- [60] J. Hindmarsh, R. Rose, A model of neuronal bursting using three coupled first order differential equations. *Proceedings of the Royal society of London. Series B. Biological sciences* **221**, 87–102 (1984).
- [61] B. Hutcheon, Y. Yarom, Resonance, oscillation and the intrinsic frequency preferences of neurons. *Trends in neurosciences* **23**, 216–222 (2000).
- [62] E. M. Izhikevich, Simple model of spiking neurons. *Neural Networks, IEEE Transactions on* **14**, 1569–1572 (2003).
- [63] E. M. Izhikevich, J. Moehlis, Dynamical systems in neuroscience: The geometry of excitability and bursting. *SIAM review* **50**, 397 (2008).
- [64] B. W. Connors, M. J. Gutnick, Intrinsic firing patterns of diverse neocortical neurons. *Trends in neurosciences* **13**, 99–104 (1990).
- [65] C. M. Gray, D. A. McCormick, Chattering cells: superficial pyramidal neurons contributing to the generation of synchronous oscillations in the visual cortex. *Science* **274**, 109–113 (1996).
- [66] J. R. Gibson, M. Beierlein, B. W. Connors, Two networks of electrically coupled inhibitory neurons in neocortex. *Nature* **402**, 75–79 (1999).
- [67] E. M. Izhikevich, G. M. Edelman, Large-scale model of mammalian thalamocortical systems. *Proceedings of the national academy of sciences* **105**, 3593–3598 (2008).

- [68] T. Binzegger, R. J. Douglas, K. A. Martin, A quantitative map of the circuit of cat primary visual cortex. *The Journal of Neuroscience* **24**, 8441–8453 (2004).
- [69] D. C. Van Essen, *et al.*, The wu-minn human connectome project: an overview. *Neuroimage* **80**, 62–79 (2013).
- [70] B. Biswal, F. Zerrin Yetkin, V. M. Haughton, J. S. Hyde, Functional connectivity in the motor cortex of resting human brain using echo-planar mri. *Magnetic resonance in medicine* **34**, 537–541 (1995).
- [71] M. Lowe, B. Mock, J. Sorenson, Functional connectivity in single and multislice echoplanar imaging using resting-state fluctuations. *Neuroimage* **7**, 119–132 (1998).
- [72] G. L. Shulman, *et al.*, Common blood flow changes across visual tasks: Ii. decreases in cerebral cortex. *Journal of cognitive neuroscience* **9**, 648–663 (1997).
- [73] M. E. Raichle, *et al.*, A default mode of brain function. *Proceedings of the National Academy of Sciences* **98**, 676–682 (2001).
- [74] D. A. Gusnard, M. E. Raichle, Searching for a baseline: functional imaging and the resting human brain. *Nature reviews. Neuroscience* **2**, 685 (2001).
- [75] M. D. Greicius, B. Krasnow, A. L. Reiss, V. Menon, Functional connectivity in the resting brain: a network analysis of the default mode hypothesis. *Proceedings of the National Academy of Sciences* **100**, 253–258 (2003).
- [76] C. F. Beckmann, M. DeLuca, J. T. Devlin, S. M. Smith, Investigations into resting-state connectivity using independent component analysis. *Philosophical Transactions of the Royal Society of London B: Biological Sciences* **360**, 1001–1013 (2005).
- [77] M. D. Fox, *et al.*, The human brain is intrinsically organized into dynamic, anticorrelated functional networks. *Proceedings of the National Academy of Sciences of the United States of America* **102**, 9673–9678 (2005).

- [78] J. Damoiseaux, *et al.*, Consistent resting-state networks across healthy subjects. *Proceedings of the national academy of sciences* **103**, 13848–13853 (2006).
- [79] M. D. Fox, A. Z. Snyder, J. L. Vincent, M. E. Raichle, Intrinsic fluctuations within cortical systems account for intertrial variability in human behavior. *Neuron* **56**, 171–184 (2007).
- [80] M. D. Fox, M. E. Raichle, Spontaneous fluctuations in brain activity observed with functional magnetic resonance imaging. *Nature reviews. Neuroscience* **8**, 700 (2007).
- [81] S. M. Smith, *et al.*, Correspondence of the brain’s functional architecture during activation and rest. *Proceedings of the National Academy of Sciences* **106**, 13040–13045 (2009).
- [82] R. L. Buckner, *et al.*, Cortical hubs revealed by intrinsic functional connectivity: mapping, assessment of stability, and relation to alzheimer’s disease. *Journal of Neuroscience* **29**, 1860–1873 (2009).
- [83] J. Wang, *et al.*, Parcellation-dependent small-world brain functional networks: a resting-state fmri study. *Human brain mapping* **30**, 1511–1523 (2009).
- [84] D. M. Cole, S. M. Smith, C. F. Beckmann, Advances and pitfalls in the analysis and interpretation of resting-state fmri data. *Frontiers in systems neuroscience* **4** (2010).
- [85] M. H. Lee, C. D. Smyser, J. S. Shimony, Resting-state fmri: a review of methods and clinical applications. *American Journal of Neuroradiology* **34**, 1866–1872 (2013).
- [86] N. Chatterjee, S. Sinha, Understanding the mind of a worm: hierarchical network structure underlying nervous system function in *c. elegans*. *Progress in brain research* **168**, 145–153 (2007).

- [87] O. Sporns, C. J. Honey, R. Kötter, Identification and classification of hubs in brain networks. *PloS one* **2**, e1049 (2007).
- [88] D. S. Bassett, *et al.*, Hierarchical organization of human cortical networks in health and schizophrenia. *Journal of Neuroscience* **28**, 9239–9248 (2008).
- [89] D. S. Modha, R. Singh, Network architecture of the long-distance pathways in the macaque brain. *Proceedings of the National Academy of Sciences* **107**, 13485–13490 (2010).
- [90] M. P. van den Heuvel, O. Sporns, Network hubs in the human brain. *Trends in cognitive sciences* **17**, 683–696 (2013).
- [91] E. Bullmore, O. Sporns, Complex brain networks: graph theoretical analysis of structural and functional systems. *Nature Reviews Neuroscience* **10**, 186–198 (2009).
- [92] D. S. Bassett, E. Bullmore, Small-world brain networks. *The neuroscientist* **12**, 512–523 (2006).
- [93] S. Achard, R. Salvador, B. Whitcher, J. Suckling, E. Bullmore, A resilient, low-frequency, small-world human brain functional network with highly connected association cortical hubs. *The Journal of Neuroscience* **26**, 63–72 (2006).
- [94] S. Sinha, J. Saramäki, K. Kaski, Emergence of self-sustained patterns in small-world excitable media. *Physical Review E* **76**, 015101 (2007).
- [95] O. Sporns, J. D. Zwi, The small world of the cerebral cortex. *Neuroinformatics* **2**, 145–162 (2004).
- [96] O. Sporns, D. R. Chialvo, M. Kaiser, C. C. Hilgetag, Organization, development and function of complex brain networks. *Trends in cognitive sciences* **8**, 418–425 (2004).
- [97] C. Haldeman, J. M. Beggs, Critical branching captures activity in living neural networks and maximizes the number of metastable states. *Physical review letters* **94**, 058101 (2005).

- [98] J. M. Beggs, Neuronal avalanche. *Scholarpedia* **2**, 1344 (2007).
- [99] A. Tang, *et al.*, A maximum entropy model applied to spatial and temporal correlations from cortical networks in vitro. *The Journal of Neuroscience* **28**, 505–518 (2008).
- [100] G. P. Moore, D. Perkel, J. Segundo, Statistical analysis and functional interpretation of neuronal spike data. *Annual review of physiology* **28**, 493–522 (1966).
- [101] D. H. Perkel, T. H. Bullock, Neural coding. *Neurosciences Research Program Bulletin* (1968).
- [102] W. Bialek, F. Rieke, R. R. d. R. van Steveninck, D. Warland, *Advances in neural information processing systems* (1990), pp. 36–43.
- [103] M. Mulansky, N. Bozanic, A. Sburlea, T. Kreuz, *Event-based Control, Communication, and Signal Processing (EBCCSP), 2015 International Conference on* (IEEE, 2015), pp. 1–8.
- [104] F. Rieke, *Spikes: exploring the neural code* (MIT press, 1999).
- [105] J.-P. Lachaux, E. Rodriguez, J. Martinerie, F. J. Varela, *et al.*, Measuring phase synchrony in brain signals. *Human brain mapping* **8**, 194–208 (1999).
- [106] M. Le Van Quyen, *et al.*, Comparison of hilbert transform and wavelet methods for the analysis of neuronal synchrony. *Journal of neuroscience methods* **111**, 83–98 (2001).
- [107] T. Kreuz, J. S. Haas, A. Morelli, H. D. Abarbanel, A. Politi, Measuring spike train synchrony. *Journal of neuroscience methods* **165**, 151–161 (2007).
- [108] T. Kreuz, D. Chicharro, R. G. Andrzejak, J. S. Haas, H. D. Abarbanel, Measuring multiple spike train synchrony. *Journal of neuroscience methods* **183**, 287–299 (2009).

- [109] D. Chicharro, T. Kreuz, R. G. Andrzejak, What can spike train distances tell us about the neural code? *Journal of neuroscience methods* **199**, 146–165 (2011).
- [110] A. Riehle, S. Grün, M. Diesmann, A. Aertsen, Spike synchronization and rate modulation differentially involved in motor cortical function. *Science* **278**, 1950–1953 (1997).
- [111] P. Fries, A mechanism for cognitive dynamics: neuronal communication through neuronal coherence. *Trends in cognitive sciences* **9**, 474–480 (2005).
- [112] U. Rutishauser, I. B. Ross, A. N. Mamelak, E. M. Schuman, Human memory strength is predicted by theta-frequency phase-locking of single neurons. *Nature* **464**, 903–907 (2010).
- [113] F. Dretske, Knowledge and the flow of information (1981).
- [114] M. Kaminski, K. J. Blinowska, A new method of the description of the information flow in the brain structures. *Biological cybernetics* **65**, 203–210 (1991).
- [115] M. Steriade, D. A. McCormick, T. J. Sejnowski, Thalamocortical oscillations in the sleeping and aroused brain. *SCIENCE-NEW YORK THEN WASHINGTON-* **262**, 679–679 (1993).
- [116] E. Salinas, T. J. Sejnowski, Correlated neuronal activity and the flow of neural information. *Nature reviews. Neuroscience* **2**, 539 (2001).
- [117] S. M. Sherman, R. Guillery, The role of the thalamus in the flow of information to the cortex. *Philosophical Transactions of the Royal Society of London B: Biological Sciences* **357**, 1695–1708 (2002).
- [118] A. Korzeniewska, M. Mańczak, M. Kamiński, K. J. Blinowska, S. Kasicki, Determination of information flow direction among brain structures by a modified directed transfer function (ddtf) method. *Journal of neuroscience methods* **125**, 195–207 (2003).

-
- [119] N. Ramnani, T. E. Behrens, W. Penny, P. M. Matthews, New approaches for exploring anatomical and functional connectivity in the human brain. *Biological psychiatry* **56**, 613–619 (2004).
- [120] R. Marois, J. Ivanoff, Capacity limits of information processing in the brain. *Trends in cognitive sciences* **9**, 296–305 (2005).
- [121] T. Paus, Mapping brain maturation and cognitive development during adolescence. *Trends in cognitive sciences* **9**, 60–68 (2005).
- [122] N. U. Dosenbach, *et al.*, Distinct brain networks for adaptive and stable task control in humans. *Proceedings of the National Academy of Sciences* **104**, 11073–11078 (2007).
- [123] M. Dhamala, G. Rangarajan, M. Ding, Analyzing information flow in brain networks with nonparametric granger causality. *Neuroimage* **41**, 354–362 (2008).
- [124] A. David, L. Pierre, *The hippocampus book* (Oxford University Press, 2009).
- [125] L. L. Colgin, *et al.*, Frequency of gamma oscillations routes flow of information in the hippocampus. *Nature* **462**, 353 (2009).
- [126] M. B. Schippers, C. Keysers, Mapping the flow of information within the putative mirror neuron system during gesture observation. *Neuroimage* **57**, 37–44 (2011).
- [127] M. Diesmann, M.-O. Gewaltig, A. Aertsen, Stable propagation of synchronous spiking in cortical neural networks. *Nature* **402**, 529–533 (1999).
- [128] M. F. Glasser, *et al.*, A multi-modal parcellation of human cerebral cortex. *Nature advance online publication* (2016). Article.
- [129] D. R. Chialvo, P. Balenzuela, D. Fraiman, The brain: what is critical about it? *arXiv preprint arXiv:0804.0032* (2008).
- [130] D. R. Chialvo, P. Bak, Learning from mistakes. *Neuroscience* **90**, 1137–1148 (1999).

- [131] J. M. Beggs, The criticality hypothesis: how local cortical networks might optimize information processing. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* **366**, 329–343 (2008).
- [132] M. G. Kitzbichler, M. L. Smith, S. R. Christensen, E. Bullmore, Broadband criticality of human brain network synchronization. *PLoS Comput Biol* **5**, e1000314 (2009).
- [133] The human brain is on the edge of chaos, <http://phys.org/news/2009-03-human-brain-edge-chaos.html>. Accessed: 2016-08-30.
- [134] The human brain is on the edge of chaos, <http://www.cam.ac.uk/research/news/the-human-brain-is-on-the-edge-of-chaos>. Accessed: 2016-08-30.
- [135] N. Bertschinger, T. Natschläger, Real-time computation at the edge of chaos in recurrent neural networks. *Neural computation* **16**, 1413–1436 (2004).
- [136] R. Legenstein, W. Maass, Edge of chaos and prediction of computational performance for neural circuit models. *Neural Networks* **20**, 323–334 (2007).
- [137] J. P. Crutchfield, K. Young, *The Santa Fe Institute, Westview* (Citeseer, 1988).
- [138] J. P. Crutchfield, K. Young, Inferring statistical complexity. *Physical Review Letters* **63**, 105 (1989).
- [139] L. Abbott, T. B. Kepler, *Statistical mechanics of neural networks* (Springer, 1990), pp. 5–18.
- [140] J. Touboul, Bifurcation analysis of a general class of nonlinear integrate-and-fire neurons. *SIAM Journal on Applied Mathematics* **68**, 1045–1079 (2008).
- [141] S. Nobukawa, H. Nishimura, T. Yamanishi, J.-Q. Liu, *Soft Computing and Intelligent Systems (SCIS), 2014 Joint 7th International Conference on*

- and Advanced Intelligent Systems (ISIS), 15th International Symposium on* (IEEE, 2014), pp. 813–818.
- [142] S. Nobukawa, H. Nishimura, T. Yamanishi, J.-Q. Liu, Analysis of chaotic resonance in izhikevich neuron model. *PloS one* **10**, e0138919 (2015).
- [143] S. Nobukawa, H. Nishimura, T. Yamanishi, J.-Q. Liu, Chaotic states induced by resetting process in izhikevich neuron model. *Journal of Artificial Intelligence and Soft Computing Research* **5**, 109–119 (2015).
- [144] R. Brette, *et al.*, Simulation of networks of spiking neurons: a review of tools and strategies. *Journal of computational neuroscience* **23**, 349–398 (2007).
- [145] H. Markram, J. Lübke, M. Frotscher, B. Sakmann, Regulation of synaptic efficacy by coincidence of postsynaptic aps and epsps. *Science* **275**, 213–215 (1997).
- [146] D. D. Lee, H. S. Seung, Learning the parts of objects by non-negative matrix factorization. *Nature* **401**, 788–791 (1999).
- [147] S. Song, K. D. Miller, L. F. Abbott, Competitive hebbian learning through spike-timing-dependent synaptic plasticity. *Nature neuroscience* **3**, 919–926 (2000).
- [148] S. J. Kuhlman, Z. J. Huang, High-resolution labeling and functional manipulation of specific neuron types in mouse brain by cre-activated viral gene expression. *PloS one* **3**, e2005 (2008).
- [149] H.-C. Tsai, *et al.*, Phasic firing in dopaminergic neurons is sufficient for behavioral conditioning. *Science* **324**, 1080–1084 (2009).
- [150] B. Tursun, T. Patel, P. Kratsios, O. Hobert, Direct conversion of c. elegans germ cells into specific neuron types. *Science* **331**, 304–308 (2011).
- [151] R. Mozzachiodi, J. H. Byrne, More than synaptic plasticity: role of nonsynaptic plasticity in learning and memory. *Trends in neurosciences* **33**, 17–26 (2010).

- [152] C. Hansel, D. J. Linden, E. D'Angelo, Beyond parallel fiber ltd: the diversity of synaptic and non-synaptic plasticity in the cerebellum. *Nature neuroscience* **4**, 467–475 (2001).
- [153] A. Roxin, H. Riecke, S. A. Solla, Self-sustained activity in a small-world network of excitable neurons. *Physical review letters* **92**, 198101 (2004).
- [154] D. Guo, C. Li, Self-sustained irregular activity in 2-d small-world networks of excitatory and inhibitory neurons. *IEEE Transactions on Neural Networks* **21**, 895–905 (2010).
- [155] D. Guo, *et al.*, Regulation of irregular neuronal firing by autaptic transmission. *Scientific reports* **6** (2016).
- [156] O. Sporns, G. Tononi, G. M. Edelman, Theoretical neuroanatomy: relating anatomical and functional connectivity in graphs and cortical connection matrices. *Cerebral cortex* **10**, 127–141 (2000).
- [157] D. J. Watts, S. H. Strogatz, Collective dynamics of small-world networks. *nature* **393**, 440–442 (1998).
- [158] M. E. Newman, D. J. Watts, Scaling and percolation in the small-world network model. *Physical Review E* **60**, 7332 (1999).
- [159] P. Bak, C. Tang, K. Wiesenfeld, Self-organized criticality: An explanation of the 1/f noise. *Physical review letters* **59**, 381 (1987).
- [160] P. Bak, C. Tang, K. Wiesenfeld, Self-organized criticality. *Physical review A* **38**, 364 (1988).
- [161] P. Bak, *How nature works* (Springer, 1996), pp. 1–32.
- [162] D. L. Turcotte, Self-organized criticality. *Reports on progress in physics* **62**, 1377 (1999).
- [163] D. P. Feldman, J. P. Crutchfield, Measures of statistical complexity: Why? *Physics Letters A* **238**, 244–252 (1998).

- [164] R. Lopez-Ruiz, H. Mancini, X. Calbet, A statistical measure of complexity. *arXiv preprint nlin/0205033* (2002).
- [165] C. Anteneodo, A. R. Plastino, Some features of the lópez-ruiz-mancini-calbet (lmc) statistical measure of complexity. *Physics Letters A* **223**, 348–354 (1996).
- [166] J. Angulo, J. Antolín, K. Sen, Fisher–shannon plane and statistical complexity of atoms. *Physics Letters A* **372**, 670–674 (2008).
- [167] M. Martin, A. Plastino, O. Rosso, Statistical complexity and disequilibrium. *Physics Letters A* **311**, 126–132 (2003).
- [168] M. Martin, A. Plastino, O. Rosso, Generalized statistical complexity measures: Geometrical and analytical properties. *Physica A: Statistical Mechanics and its Applications* **369**, 439–462 (2006).
- [169] S. Rani, G. Sikka, Recent techniques of clustering of time series data: a survey. *International Journal of Computer Applications* **52** (2012).
- [170] S. Aghabozorgi, A. S. Shirkhorshidi, T. Y. Wah, Time-series clustering—a decade review. *Information Systems* **53**, 16–38 (2015).
- [171] Hierarchical clustering, <https://www.mathworks.com/help/stats/hierarchical-clustering.html>. Accessed: 2017-07-11.
- [172] G. Punj, D. W. Stewart, Cluster analysis in marketing research: Review and suggestions for application. *Journal of marketing research* pp. 134–148 (1983).
- [173] D. J. Ketchen Jr, C. L. Shook, The application of cluster analysis in strategic management research: an analysis and critique. *Strategic management journal* pp. 441–458 (1996).
- [174] G. W. Milligan, M. C. Cooper, An examination of procedures for determining the number of clusters in a data set. *Psychometrika* **50**, 159–179 (1985).

- [175] B. K. Teh, S. A. Cheong, Cluster fusion-fission dynamics in the singapore stock exchange. *The European Physical Journal B* **88**, 263 (2015).
- [176] C.-S. J. Chu, Time series segmentation: A sliding window approach. *Information Sciences* **85**, 147–173 (1995).
- [177] E. Zivot, J. Wang, *Modeling Financial Time Series with S-Plus®* (Springer, 2003), pp. 299–346.
- [178] M. Rosvall, C. T. Bergstrom, Mapping change in large networks. *PloS one* **5**, e8694 (2010).
- [179] Source code for verkko.plots.alluvial, http://becs.aalto.fi/~darstr1/verkkko/docs/build/html/_modules/verkkko/plots/alluvial.html. Accessed: 2017-07-11.
- [180] W. M. Rand, Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical association* **66**, 846–850 (1971).
- [181] L. Hubert, P. Arabie, Comparing partitions. *Journal of classification* **2**, 193–218 (1985).
- [182] P. Dayan, L. F. Abbott, *Theoretical neuroscience*, vol. 806 (Cambridge, MA: MIT Press, 2001).