Alma Mater Studiorum · University of Bologna

School of Science
Department of Physics and Astronomy
Master Degree in Physics

Synchronisation Phenomena in Complex Neuronal Networks

Supervisor:
Prof. Armando Bazzani

Submitted by:
Giulio Colombini

Academic Year 2020/21

Nec me animi fallit Graiorum obscura reperta
difficile inlustrare Latinis versibus esse,
multa novis verbis praesertim cum sit agendum
propter egestatem linguae et rerum novitatem;
sed tua me virtus tamen et sperata voluptas
suavis amicitiae quemvis efferre laborem
suadet et inducit noctes vigilare serenas
quaerentem dictis quibus et quo carmine demum
clara tuae possim praepandere lumina menti,
res quibus occultas penitus convisere possis.

Titus Lucretius Carus, *De Rerum Natura*,
Liber Primus, 136-145

To those who taught me curiosity.

Abstract

The phenomenon of neural synchronisation, a simultaneous and repeated firing of clusters of neurons, underlies many physiological functions and pathological manifestations in the brain of humans and animals, ranging from information encoding to epileptic seizures. Neural synchronisation, as a general phenomenon, can be approached theoretically in the framework of Dynamical Systems on Networks. In the present work, we do so by considering complex networks of FitzHugh-Nagumo model neurons. In the first part we consider the most understood models where each neuron treats its presynaptic neurons all on an equal footing, normalising signals with its in-degree. We study the stability of the synchronous state by devising an algorithm that destabilises it by selecting and removing links from the network, so to obtain a bipartite network. The selection is performed using a perturbative expression, which can be regarded as a specialisation of a previously introduced Spectral Centrality measure. The algorithm is tested on Erdős-Renyi, Watts-Strogatz and Barabási-Albert networks, and its behaviour is assessed from a dynamical and from a structural point of view. In the second part we consider the less studied case in which each neuron divides equally its output among the postsynaptic neurons, so to reproduce schematically the situation where a fixed quantity of neurotransmitter is subdivided between several efferent neurons. In this context a self-consistent approach is formulated and its limitations are explored. In order to extend its application to larger networks, a Mean Field Approximation is presented. The predictivity of the Mean Field Approach is then tested on the different random network models, and the results are discussed in terms of the original network properties.

Contents

Introduction

The neuron doctrine, one of the founding elements of modern neuroscience, can be considered to begin in the late 19$^{\text{th}}$ century, with the discovery by Santiago Ramón y Cajal, Figure 1, that the nervous tissue is not built as a single reticulum, but rather as a network of elementary units, among which neurons play the most prominent role. Indeed neurons are not the only cellular component, possibly not even the most abundant [BBH16], within the nervous system: glia cells, for example, play a relevant structural and protective role, and some specialised ones, called astrocytes, are being investigated for their possible role in the formation of memory engrams. The peculiarity of neurons, although, rests in their rich electrophysiological features, which make them active electrical elements, capable of information processing and storage, when combined in a network. These premises make the concept of network a fundamental model, when studying collective neural phenomena.

One could summarise the concept of network as that of a set of individual, elementary objects, called the *nodes*, together with the set of their pairwise interactions or relationships, called *edges* or *links* [New18]. A great number of tools, constituting the field of *network theory* has been developed to characterise these objects from various standpoints, such as combinatorics, stochastic processes or algebra. Although not having been formulated explicitly for the study of neuroscientific phenomena, this framework is highly suited to their understanding, and even if the identification is often made between nodes and neurons, in recent times network models have been applied successfully also at a variety of larger and smaller scales, ranging from representations of interactions at a cortical level to the modelling of exchanges at the intra-cellular scale [BZG18]. Furthermore, the *nature* of the relationships represented by the links can vary greatly: representing synapses, but also functional interdependence, spatio-temporal correlation in data or physical proximity. In addition, a degree of heterogeneity and dynamics can be inserted, considering multilayer or time evolving links and nodes [Leh+14]. A review with examples for all these types of models can be found in [BZG18], where an approach is proposed to understand and appreciate their differences, which is based on three figurative

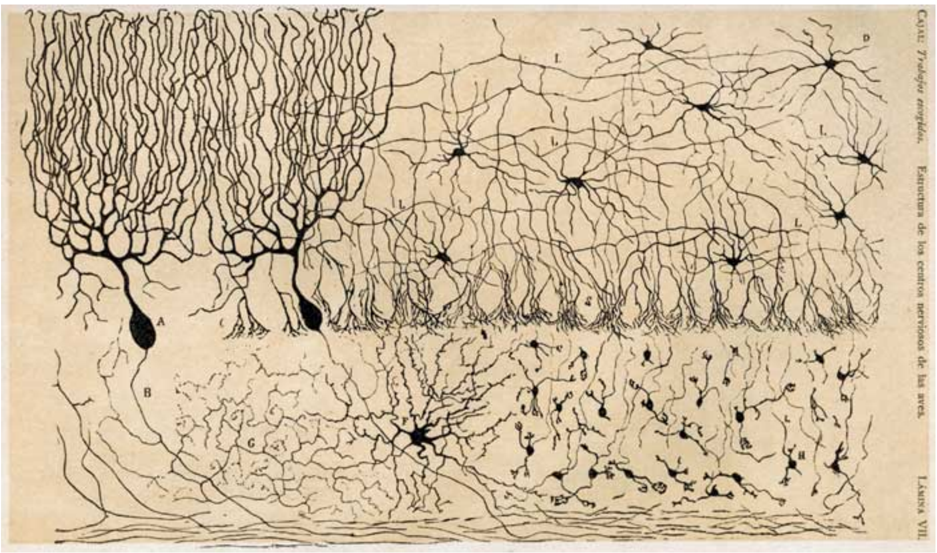
CAJAL: Trabajos escogidos. Estructura de los centros nerviosos de las aves.
LÁMINA VII.
A
B
C
D
G
H
L

Figure 1: One of the first microscope drawings by Ramón y Cajal, performed with the Golgi silver staining method. Picture from *Estructura de los centros nervosos de las aves*, Madrid 1905. The samples are from a cerebellum of *Gallus gallus domesticus* (chicken).

cartesian axes.

The first one is the *data representation vs first principles* axis. On one end we find detailed network representation of some experimentally obtained data, whereas on the other we find theoretical networks of dynamical elements, attempting to explain the mechanism behind some observed behaviours. The drawbacks of the first approach are that data representations are not very suited to predictions, while the first principles models tend to simplify many physiological features.

Then, models can be cathegorised by the focus being on *biophysical realism vs functional phenomenology*. This spans from the case where links represent actual physical elements, such as axons, to the case where they represent functional interdependence. The former have the advantage of incorporating many physiological features, but can prove computationally very expensive, or difficult to interpret, due to the high number of parameters needed to describe the network. Conversely the latter can be used to investigate the informational nature of the brain processes, but are harder to map onto the actual structure of the brain.

Finally, the third axis represents the *elementary vs coarse grained* nature of the model. Indeed, the network approach is based on the identification of well defined discrete units, therefore the graining one chooses is a relevant feature of the model. Models where each node represents a single neuron are better suited to studies on neural encoding or on cellular function on a small scale, whereas by using a coarser graining larger brain models can be studied.

In the present work we will be considering on each network node a reasonably realistic mesoscale dynamical model suited to represent a neuron, or a small cluster of neurons, with its electrophysiological properties. From the point of view of network structure, we will be considering artificial random network architectures, with well understood properties, so to focus our attention on the study of collective neuronal synchronisation phenomena.

In the Neurosciences, synchronisation phenomena are observed at a variety of scales, ranging from single neurons, to small ensembles and even at the scale of entire cortical regions. Synchronisation is believed to play a fundamental role both in physiological functions (sleep-wake cycle, information encoding) [Sin99], and pathological conditions (Parkinson's disease, epileptic seizures [Ger+20]).

The first scientific approach to the study of the synchronisation phenomenon dates back to the observation by Christiaan Huygens of the synchronised motion of two pendulum clocks, when both were hanging from the same wooden beam, Figure 2. He observed that, after roughly thirty minutes from their starting, the clocks started oscillating with equal frequency, but opposed phases. Since at the time, the term "synchronisation" was not yet in use, he referred to the phenomenon as to the development of a "sympathy" between the two pendula. In a letter to his father, written in 1665 [Huy99], he was the first to attribute the cause of synchronisation to the small interaction provided by the wooden beam, having carefully ruled out experimentally his previous hypothesis, that the synchronisation was due to air currents displaced by the oscillating motions. The first theory for the synchronisation of nonlinear systems dates, instead, to the third decade of the 20th century, and is due to Balthasar Van der Pol. Due to their great relevance in the radio systems of the time, Van der Pol was studying the synchronisation of some electrical oscillations that take place in vacuum tubes, when they are subject to an external forcing [VV27]. The study of the synchronisation of periodic oscillations for low dimensional systems, has then been brought forward by the russian school of mathematics, with contributions from authors such as Aleksandr Andronov [AW30], Vladimir Arnol'd and Arkady Pikovsky [PRK01].

Oftentimes, although, in Nature we observe the synchronisation of a macroscopic number of systems, displaying so called *collective synchronisation*. The first approach to this problem has been provided by Arthur Winfree [Win67], and Yoshiki Kuramoto [Kur75], who studied the syncronisation phase transition for networks of coupled phase oscillators, very simple systems with a single periodic dynamical variable. Thanks to the simplicity of the elementary sys-

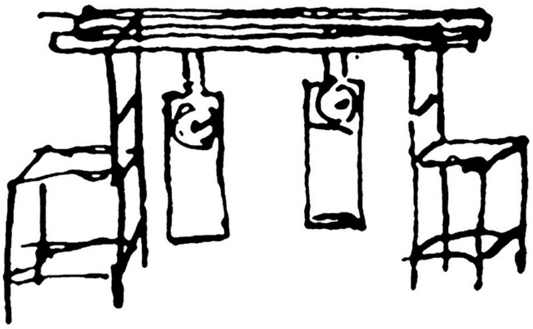

Figure 2: Sketch by Christiaan Huygens representing the synchronisation of two clocks, from the letter to his father where he first reported the phenomenon. Two pendulum clocks can be seen suspended from a wooden beam, which in turn is held above ground by two chairs.

tem, researchers have been able to characterise quite well the synchronisation transition that appears in the Kuramoto model. For more complicated node dynamics, the determinant contribution, which will be often used in the present work, has been made in 1998 by Louis Pecora and Thomas Carroll [PC98], with the introduction of the Master Stability Function. This framework allows to simplify greatly the assessment of linear stability for synchronised solutions, by decoupling the dynamical component from the network structure. The formalism was initially designed to study the stability of complete synchronisation for networks of chaotic systems, but is very useful in general when considering networks of nonlinear dynamical systems, such as those that try to replicate the behaviour of neurons, and can be generalised to deal with delay-coupled systems.

In this dissertation we will concern ourselves with the study of synchronisation phenomena for a network of FitzHugh-Nagumo neurons, so to understand their relationship with the network structure. In Chapter 1, we introduce the concept of dynamical network model, along with some fundamental elements of network theory. We also look at the electrophysiology of the neuron, so to introduce the neuronal model of choice. Finally we introduce the neuronal network model we shall consider, in the generic mathematical framework of dynamical systems on graphs. In Chapter 2 we review the most studied coupling scheme for FitzHugh-Nagumo neurons, and propose an algorithm that is able to destabilise the completely synchronous state of a network, by introducing a bipartite stable state. This is achieved by selecting and cutting network links on the basis of the change they induce in spectral and structural properties of the network. A link is established between the expressions used to predict the changes in the spectrum and the previously introduced concept of k-spectral centrality. In Chapter 3, we consider a scheme less investigated in previous literature, and a mean connectivity approach is proposed, to obtain an upper bound on the minimal connection strength needed for the network to synchronise. The approach is then shown, by simulation, to work best on random networks with highly regular connectivity, such as Erdős-Renyi models, and to fail when one considers networks with highly spread degree distributions, such as scale-free models.

In the Conclusion we summarise the original results obtained in the dissertation. In the context of the first normalisation scheme, we are able to devise an algorithm that destabilises the completely synchronous state, by bipartitioning the network. The algorithm, although, tends to destroy the structural properties on both small-world and scale-free architectures. Future development in this area can be aimed at the overcoming of such problems, as it would be desirable, of algorithms of this type, not to erase such relevant network properties in the destabilisation process. As far as the second scheme is concerned, we are able to prove that on the Erdős-Renyi architecture, the quality of the Mean Field Approximation improves with the network size, by observing the onset of the Central Limit Theorem. Conversely, for the Barabási-Albert model, we show that the Mean Field Approximation fails due to the non-negligibility of connectivity fluctuations. Further development of this approach should focus both on the numerical simulation of larger systems, so to study the actual behaviour in the limit of large networks, and on a theoretical development and contextualisation of these synchronisation transitions in out-of-equilibrium Statistical Mechanics.

Acknowledgements

We would like to thank Professor Nicola Guglielmi for providing the original `FORTRAN 90` codebase for the DDE integration algorithm `RADAR 5` (see Appendix C), which has proved fundamental in the development of the present work.

Chapter 1

Dynamical Model

In this Chapter we first review the concept of dynamical network model, and introduce some Network theory elements, then we present the properties of the neuronal model considered in this dissertation, with attention to their relationship with the electrophysiological properties of real neurons. Finally, we look at the coupling schemes that we will consider in the work.

1.1 Dynamical models on graphs

When considering systems where elementary units can be individuated, that can interchange signals or information between one another, it is natural to formulate the problem in terms of dynamical network models. In these models, the units, identified with nodes of the network, are treated as dynamical systems, endowed with a phase space and internal dynamics of their own. This dynamics can take a host of forms, depending on the phenomenon under study, and can be linear, non-linear, chaotic or even encompass some degree of stochasticity, such as a random forcing. The interaction pattern of each unit is specified by the adjacency structure of the network, and the actual dynamical form of the interaction will be specified by a function, called the coupling function, that can possibly include different forms of delay to account for propagation or processing times. The network structure, in some cases, can become a dynamical entity in itself and change over time. This picture translates mathematically into the general dynamical equation

$$\dot{\mathbf{x}}_i(t) = \mathbf{f}\left(t, \mathbf{x}_i(t), \boldsymbol{\xi}_i(t)\right) + \sum_{j=1}^{N} A_{ij}(t)\mathbf{g}\left(\mathbf{x}_i(t), \mathbf{x}_j(t-\tau)\right) \tag{1.1}$$

where $i, j \in 1, \ldots, N$ label the network nodes. Each node is endowed with a d-dimensional state $\mathbf{x}_i \in \mathbb{R}^d$, and the local dynamics is specified by $\mathbf{f}$, a function of the current state, time t to account for possible external forcing and possibly a stochastic variable $\boldsymbol{\xi}_i(t)$ implementing noise. The coupling function is given by $\mathbf{g}$. Notice how the possibility of A, the adjacency matrix of the underlying network, being time dependent, is encompassed.

This kind of model has been successfully employed to understand many different phenomena, from ecological interactions between species [Sto+02], to, more recently, epidemics spreading [Del+20]. A field where these models have been quite successful is Neuroscience, where networks of model neurons have been able to reproduce complex phenomena, such as epileptic seizures-like manifestations [Ger+20], or unihemispheric sleep [Ram+19]. Both of these are actually synchronisation-related phenomena, and the progress in understanding them via network models is an additional incentive to pursue such research.

To adapt Equation (1.1) to the study of neural phenomena, it is necessary to select some node dynamics and coupling function that depend on the scale of the model, and kind of relationship highlighted by the network structure. For example, at the microscale, each node could encompass a full Hodgkin-Huxley dynamics [HH52], so to describe a single neuron. As one moves towards single nodes representing small neuron ensembles or cortical patches, the model of choice often becomes the FitzHugh-Nagumo [Fit61], [NAY62], or Hindmarsh-Rose [Izh06] model. For even larger scales, acceptable results have been obtained also with simple Kuramoto phase oscillators. As far as the coupling functions are concerned, the choice at the microscale can be made among various synapse models [Rot13]. At larger scales usually some form of, possibly delayed, diffusive coupling is used, on grounds of mathematical practicality and simplicity of treatment.

1.2 Basic concepts in Network Theory

In this section, we review some fundamental concepts in network theory, which will be useful in the following. We start by defining a network, or graph, as follows

Definition 1.2.1 (Graph). We define *Graph* a pair $G(V, E)$, where V and E are sets, and where the elements of V are called *nodes* or *vertices*, and the elements of E are pairs of vertices and are called *edges* or *links*.

In general we will be denoting by N the number of nodes in a graph, and label without loss of generality the nodes with the integers from 1 to N. Two nodes that are linked by an edge are termed *adjacent*. Additionally to this definition, whenever a graph has at most one edge connecting any pair of nodes, and has no self-edges, i.e. no edges that link a node to itself, we will refer to it as a *simple graph*.

The Adjacency Matrix There are a number of ways to create a mathematical *representation* of a network, the first, and fundamental one, is the *Adjacency Matrix*.

Definition 1.2.2 (Adjacency matrix). Let $G(V, E)$ be a graph. We call the *adjacency matrix* of G the $N \times N$ matrix built as follows

$$A_{ij} = \begin{cases} 1 \text{ if there is an edge connecting node } j \text{ to node } i \\ 0 \text{ otherwise} \end{cases} \tag{1.2}$$

notice that this definition allows us to introduce a concept of link directionality, where j can be connected to i, but the opposite may not be true. Those networks for which links have a directionality are called *directed*, whereas those for which links are always bidirectional are called *undirected* networks. With the present definitions in place, it becomes clear that undirected networks are associated to *symmetric* adjacency matrices, while adjacency matrices associated to directed networks lose this property. A very important property of a node is its *degree*

Definition 1.2.3 (Degree). Let $G(V, E)$ be an undirected graph, and let $v_i \in V$ be a node belonging to it. The *degree* of node v_i is the number of links connected to it, and for an undirected graph it can be calculated as

$$d(v_i) = \sum_{j=1}^{N} A_{ij} = \sum_{j=1}^{N} A_{ji} \tag{1.3}$$

since for undirected networks $A_{ij} = A_{ji}$. As a shorthand, on undirected graphs, we will indicate $d(v_i)$ as d_i.

On *directed* networks, we distinguish the *in-degree* and the *out-degree* of node v_i, introducing them respectively as the number of links incoming and outgoing on node v_i. These quantities can be written as

$$d_{in}(v_i) = \sum_{j=1}^{N} A_{ij} \qquad d_{out}(v_i) = \sum_{j=1}^{N} A_{ji} \tag{1.4}$$

as the adjacency matrix is no longer symmetric. Of course for undirected graphs, $d_{in} = d_{out}$ for all nodes.

For undirected networks we can build the so called *Degree Matrix*, a diagonal matrix, which contains as the i-th element on the diagonal the degree of the i-th node.

$$D = \operatorname{diag}\{d_1, d_2, \dots, d_N\} \tag{1.5}$$

We now introduce some notions on sequences of nodes and edges, which informally correspond to the possible ways to move on a graph.[1]

Definition 1.2.4 (Walk). Let $G(V, E)$ be a graph. We define *Walk* an edge sequence $(e_1, \dots, e_n)$ where $\forall i \in \{1, \dots, n-1\}$ we have $e_i = (v_i, v_{i+1}) \in E$, i.e. v_i and v_{i+1} are adjacent. If $v_n = v_1$ the walk is called *closed*, otherwise it is called open.

Notice that in the definition of walk we are being as general as possible, as both the edges and the vertices can be repeated at any time along the walk. We introduce some more specific definition in this regard.

Definition 1.2.5 (Trail, Path, Cycle). We define as a *Trail*, a Walk with no repeated links. We define as a *Path*, a Trail with no repeated nodes. We define a *Cycle*, as a closed Path.

We use the idea of Path to introduce a notion that will be useful in the following, the one of connectedness, and of components of a network.

Definition 1.2.6 (Connected Component). Let $C \subseteq V$ be a subset of the set of nodes V of $G(V, E)$. If $\forall c_1 \neq c_2 \in C$ there exists a path connecting them, and if no other nodes from V can be added to C, while preserving this property, we call C a *Connected Component* of graph G.

If within a network there exists a single connected component, then the whole network is called *connected*. Connectedness of the network is one of the properties that can be verified through the adjacency matrix. Indeed, for a network with k components, there exists an $N \times N$ permutation matrix $\mathcal{P}$ such that A is cast in a block-diagonal form

$$\mathcal{P} A \mathcal{P}^T = \begin{pmatrix} A_1 & 0 & \dots & 0 \\ 0 & A_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & A_k \end{pmatrix} \tag{1.6}$$

[1] We introduce these concepts only for undirected graphs, with which we deal in this dissertation. There exist similar concepts for directed graphs, with the necessary modifications due to the directed nature of the links.

where each of the A_l is the adjacency matrix relative to the l-th component. Consequently, connected networks, are those for which no permutation matrix $\mathcal{P}$ exists, that can cast the adjacency matrix in block diagonal form. Of course this approach is not particularly performant algorithmically, as it would entail, in the worst case, trying all permutation matrices of N elements, but it is worth mentioning as an example of the significance of the adjacency matrix.

Another class of networks, which will play a very relevant role in this work, contains those networks which allow connections only between two specific vertex subsets.

Definition 1.2.7 (Bipartite Graph). Let $G(V, E)$ be a graph. Let $P \subset V, Q = V \setminus P \subset V$ be two subsets of nodes of the graph. We say that G is bipartite if the edges of G connect only nodes belonging to different classes, i.e. have the form $e = (p, q)$ with $p \in P, q \in Q$.

The adjacency matrix can tell us something also about the bipartition of a graph, by a similar argument to the one just considered for connectedness. Indeed if we consider an undirected bipartite network, there must exist a permutation matrix $\mathcal{P}$, such that

$$\mathcal{P}A\mathcal{P}^T = \begin{pmatrix} 0 & A_b \\ A_b^T & 0 \end{pmatrix} \tag{1.7}$$

where A_b is an $n \times m$ matrix, n is the number of nodes in the first component, and $m = N - n$ the number of nodes in the second one. This is true, since via a permutation of the nodes we must be able to highlight the fact that the connections take place only *between* the two different subsets of nodes, while the zeroes on the diagonal show that there are no connections within a single partition. The adjacency matrix is a powerful tool in network theory, that can give much more information than what we have summarised so far: for a deeper presentation, the interested reader is invited to consult [New18], where these topics are presented in full detail. Now we introduce a new type of matrix, to be used in the characterisation of networks.

The Laplacian Matrix Let us consider an undirected network, with adjacency matrix A and degree matrix D. The Laplacian matrix is defined as

$$L = D - A \tag{1.8}$$

it is so called due to its relationship to diffusion processes on networks (see Appendix A for an example). Notice that the Laplacian is always a singular matrix, as it accepts the eigenvector $\mathbf{1}_N$ in its kernel by construction. Moreover, due to its symmetry, the spectrum is completely real and non-negative. The Laplacian can be normalised, the most common ways to do so are

$$L_{RW} = D^{-1}(D - A) \qquad = I - D^{-1}A \tag{1.9}$$

$$L_{symm} = D^{-1/2}(D - A)\,D^{-1/2} \qquad = I - D^{-1/2}AD^{-1/2} \tag{1.10}$$

the first is generally referred to as the *random-walk-normalised* Laplacian, while the second one is called the *symmetrically normalised* Laplacian. Notice that the two are similar matrices, as

$$D^{-1/2}L_{symm}D^{1/2} = L_{RW} \tag{1.11}$$

and therefore they share the same spectrum. In the following, we will mostly work with the random-walk-normalised Laplacian.

One of the reasons to introduce the Laplacian is the detailed characterisation of the graph that can be done through its spectrum. We will now state some known theorems about the spectrum of the normalised Laplacian, in relation to the network properties. The interested reader can find their proof and an in-depth analysis of Spectral Graph Theory in [Chu97]. In the following we shall denote as $\{\lambda_1 = 0, \lambda_2, \ldots, \lambda_N\}$ the sorted spectrum of the normalised Laplacian.

Theorem 1.2.1 (Fiedler value). Let $G(V, E)$ be a graph, and L_{RW} its associated random-walk-normalised Laplacian, with spectrum $\{\lambda_1 = 0, \lambda_2, \ldots, \lambda_N\}$. Then G is connected if and only if $\lambda_2 > 0$. Moreover, if $\lambda_1 = \lambda_2 = \ldots = \lambda_i = 0$ and $\lambda_{i+1} > 0$, G has i connected components.

Due to its role in indicating connectedness or disconnectedness λ_2 is generally referred to as *algebraic connectivity* or *Fiedler value*.

Theorem 1.2.2 (Bipartition). Let $G(V, E)$ be a graph, and L_{RW} its associated random-walk-normalised Laplacian, with spectrum $\{\lambda_1 = 0, \lambda_2, \ldots, \lambda_N\}$. Then $\lambda_i \leq 2\,\forall i$, and in particular $\lambda_N = 2$ if and only if a component of G is bipartite and non trivial.

Therefore, the spectrum of the normalised Laplacian is informative also with respect to bipartition. Moreover, bipartition entails another spectral property

Theorem 1.2.3 (Bipartition $\leftrightarrow$ symmetrical spectrum). Let $G(V, E)$ be a bipartite graph, and L_{RW} its associated random-walk-normalised Laplacian, with spectrum $\{\lambda_1 = 0, \lambda_2, \ldots, \lambda_N = 2\}$. Then for each λ_i eigenvalue of L_{RW}, also $2 - \lambda_i$ is an eigenvalue of L_{RW}.

So if the network is bipartite, we gain a further constraint on the spectrum, which must be symmetric with respect to 1.

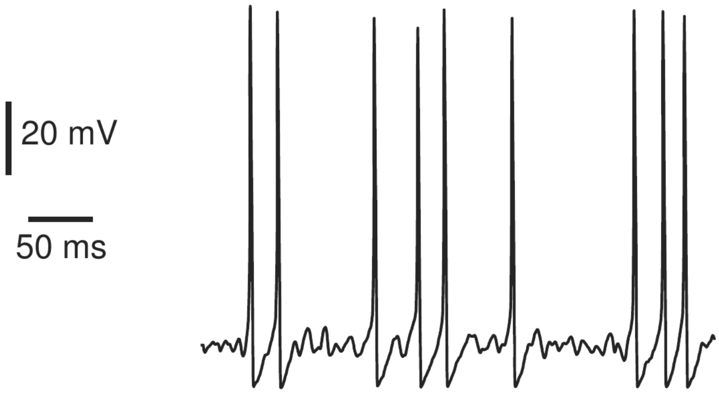

Figure 1.3.1: Membrane potential time series for a mesencephalic neuron subject to noisy input. Image from [Izh06].

1.3 The FitzHugh-Nagumo model neuron

The most relevant electrophysiological feature of the neuron is its excitability, i.e. its ability to quickly depolarise and repolarise with respect to its resting membrane potential, producing a sudden spike known as *action potential*, Figure 1.3.1, in response to electrical and chemical changes in the surrounding extracellular medium. By membrane potential we intend the difference in electric potential between the inside and the outside of the cell membrane, i.e. the extra and intracellular medium. We summarise briefly the electrophysiological mechanisms behind such a phenomenon, following the presentation of [Izh06], to which the interested reader is directed for a detailed explanation of the dynamical aspects of Neurophysiology. The currents that arise in the electrophysiology of the neuron are of ionic nature, the charge carriers being mainly Na^+, K^+, $Ca^{++}$and Cl^-. The most present in the intra-cellular medium are $K^+$and some negatively charged macromolecules A^-, while the most concentrated out of the cell are Ca^{++}, Na^+, Cl^-. The cellular membrane that separates the intra-cellular from the extra-cellular medium is a double layer of phospholipidic molecules that is impermeable to ions, but it presents some proteic channels, that allow the passage of the mentioned ions, according to their electrical potential and concentration gradients, except for the A^-, which are large molecules and cannot pass through. Thus, in principle, the ions can redistribute themselves across the membrane to reduce the concentration asymmetry between the intra-cellular and extra-cellular medium, in practice this is not always true, and the only two currents that significatively flow due to this mechanism alone are those of $K^+$and Cl^-. Furthermore, even for these two species, the concentration gradient current is not sufficient to remove the concentration asymmetry, due to several mechanisms that prevent this from happening, among which we identify the most relevant:

- Passive redistribution: the A$^-$, which are confined within the cell due to their size, attract K$^+$and repel Cl$^-$, thus contributing to the preservation of asymmetry.

- Active transport: across the cell membrane, active *ion pumps* are present, such as the Na$^+$-K$^+$pump, that brings within the cell two K$^+$for each three Na$^+$that it releases in the extracellular medium, thus maintaining the concentration of K$^+$higher inside the cell.

Were we to consider these mechanisms alone to maintain the concentration gradient, the channel current caused by the concentration asymmetry would keep flowing. On the other hand, we must take into account that the ions carry electric charges, considering for example the species K$^+$, this means that we are slowly building up positive charge outside the channel, and negative charge inside due to the unmatched A$^-$, thus giving rise to a potential difference between the interior and the exterior of the cell. At some point an equilibrium is achieved, and the concentration current is balanced out by the one caused by the potential difference. The potential difference value for which this happens is called the *Nernst equilibrium potential,* and depends on the ionic species under consideration, and environmental variables such as the temperature. In this picture, denoting E_S the Nernst potential for a species S, the current of ion species S caused by a total membrane potential difference V will be given by

$$I_S = g_S\,(V - E_S) \tag{1.12}$$

where g_S is the conductance of the ion channel associated to species S. We will comment soon more specifically on the form of this conductance. Now, if we consider Equation (1.12), for the four major ion species mentioned before, and treat the impermeable membrane as a capacitor of capacitance C [2] we are able to formulate an *equivalent circuit model* for the membrane.

$$C\dot{V} = I - g_{Na}(V - E_{Na}) - g_{Ca}(V - E_{Ca}) - g_{Cl}(V - E_{Cl}) - g_K(V - E_K) \tag{1.13}$$

where I is the total current across the membrane. Now, if all the channels were *Ohmic*, i.e. if the conductances were all constants independent from the other variables, for $I = 0$ the system would simply relax to a weighted average of the Nernst potentials

$$V_{rest} = \frac{g_{Na}E_{Na} + g_{Ca}E_{Ca} + g_{Cl}E_{Cl} + g_K E_K}{g_{Na} + g_{Ca} + g_{Cl} + g_K} \tag{1.14}$$

[2] An average value for the specific membrane capacitance per membrane surface unit is given as $0.90 \pm 0.03\,\mu$F/cm^2 for neurons in [GSC00].

which for $I \neq 0$ is simply displaced

$$V_{rest} \to V_{rest} + \frac{I}{g_{Na} + g_{Ca} + g_{Cl} + g_K} \tag{1.15}$$

This behaviour is too simple, with respect to the phenomenology we observe in neurons. Therefore, some key feature must be missing, to produce the typical spiking of neural cells. This feature comes from the fact that, with the exception of some K^+or Cl^-channels, the ion membrane channels are *not* Ohmic. The channel conductances are in general nonlinear, and depend on a number of other variables.

This phenomenon is called *channel gating*. The ionic channels are large proteins, with acqueous pores that allow for the transit of ions. Attached to these channels, although, there can be *gating particles*, that depending on some conditions of the surrounding medium, called *gating variables*, can open or close the channel. The most common gating mechanisms are

- Voltage gating: the opening or closing of a gate depends on the membrane potential → K^+, Na^+channels

- Intracellular agent gating: the gating variable for a species can be the concentration of another ionic species → Ca^{++}-gated K^+channels

- Extracellular agent (neurotransmitter) gating: opening and closing can be triggered by some neurotransmitter substance in the extracellular medium → γ-Aminobutyric acid (GABA), N-Methyl-D-aspartic acid (NMDA) gated channels

This being said, the study of the dynamics of gating is a very rich research field in itself: generally the phenomenon is approached as probabilistic in nature, with the activation and inactivation probabilities being made to evolve over time in function of the relevant gating variables.

One of the most important models that takes into account gated channels to reconstruct the electrophysiology of the neuron, is the Hodgkin-Huxley model [HH52]. It has first been proposed as a model of the Squid Giant axon, but it is widely considered representative of the general dynamics of neurons. Via experimental observations Hodgkin and Huxley determined that the relevant ionic channels and gating mechanisms for the system are

- A voltage-gated K^+current I_K, with four activation gates,

- A voltage-gated Na$^+$current I_{Na}, with three activation and one inactivation[3] gate,
- An Ohmic leak current I_L, mostly carried by Cl$^-$ions.

from which the resulting dynamical equations are

$$\begin{aligned}
C\dot{V} &= I - \overbrace{\bar{g}_K n^4 \left(V - E_K\right)}^{I_K} - \overbrace{\bar{g}_{Na} m^3 h \left(V - E_{Na}\right)}^{I_{Na}} - \overbrace{\bar{g}_L \left(V - E_L\right)}^{I_L} \\
\dot{n} &= \alpha_n(V)(1-n) + \beta_n(V)n \\
\dot{m} &= \alpha_m(V)(1-m) + \beta_m(V)m \\
\dot{h} &= \alpha_h(V)(1-h) + \beta_h(V)h
\end{aligned} \tag{1.16}$$

where the n, m and h variables are respectively the activation probability for the K$^+$channel, and the activation and inactivation probabilities for the Na$^+$channel. Many specimen of each of the three channel types are present in the membrane, with a consequent high number of gates, so that the probabilities can be interpreted in a frequentist sense, as fractions of activated or inactivated gates. The $\alpha(V)$ and $\beta(V)$ functions are respectively sigmoidal and exponential functions that describe the transition rates, and contain numerical constants that depend on the choice of the origin of potentials and on the physical properties of the specific system under consideration.

Although being very sound from a physiological point of view, the Hodgkin-Huxley model has several practical drawbacks. Due to its highly nonlinear dynamics, it can exhibit many different and possibly chaotic behaviours, which are quite difficult to study in four dimensions, without the help, for example, of the phase plane methods that can be used for nonlinear two-dimensional systems. For these reasons, in the sixties, Richard FitzHugh and Jinichi Nagumo, developed independently the so called FitzHugh-Nagumo model [Fit61], [NAY62] for neural excitability.

The FitzHugh-Nagumo model can be regarded either as a 2D projection of the Hodgkin-Huxley model, as recognised by the author in the original paper [Fit61], or as a modified version of the Van Der Pol oscillator, to account for excitable behaviour. Its dynamics is specified by a pair of nonlinear Ordinary Differential Equations, characterised by the presence of two time scales. Many formulations have been proposed, among which we choose, for simplicity

$$\begin{aligned}
\varepsilon\dot{u}(t) &= u(t) - \frac{u^3(t)}{3} - v(t) + I_{ext}(t) \\
\dot{v}(t) &= u(t) + a
\end{aligned} \tag{1.17}$$

[3] A closed and an inactivated channel correspond to different situations: in the first case, an activation gate simply did not open, while in the second an inactivation gate has closed a channel previously opened by an activation gate.

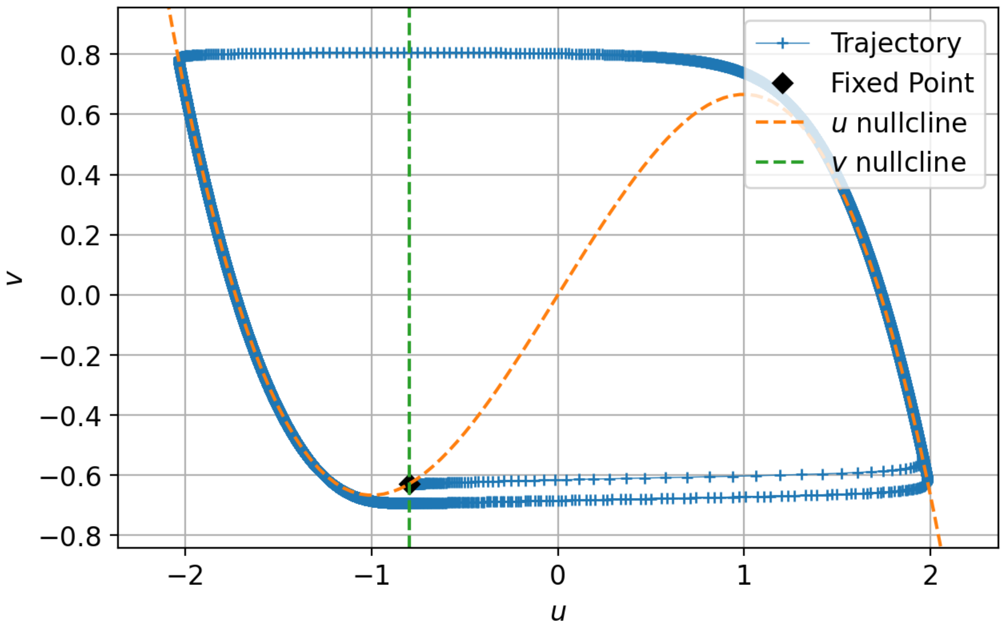

Figure 1.3.2: Phase space plot for the FHN system in the tonic spiking phase ($a = 0.8$, $\varepsilon = 0.01$).

Where u is the fast variable, presenting a quasi-threshold spiking dynamics, akin to the membrane potential. Conversely, the slow variable v is a recovery variable, representing the characteristic refractoriness of neurons after firing. The separation between the two time scales is dictated by the factor ε. I_{ext} is an external stimulus, possibly representing currents given by the interactions with other neurons, or by external stimuli. Finally, a is a dynamical parameter regulating the dynamical regime of the model. Indeed, the system possesses a single fixed point $(u^*, v^*)^T = \left(-a, -a + \frac{a^3}{3}\right)^T$ that is stable for $|a| > 1$ and becomes unstable for $|a| < 1$, when the system undergoes a supercritical Hopf Bifurcation and a stable limit cycle appears.

The first dynamical regime is termed *excitable*, and can be interpreted physiologically as that of a standard neuron, whereas the second is called a *tonic spiking* state, and can be ideally interpreted as a pacemaker neuron. To understand the mechanisms of spike production in the FHN system we can analyse its Phase Space (Figures 1.3.2, 1.3.3). We draw the *nullclines*, the curves along which the time derivative of one of the variables is zero. The v-nullcline takes the form of a vertical line at $u = -a$, while the u-nullcline is a cubic curve, intersecting the u axis at $-\sqrt{3}$, 0 and $\sqrt{3}$, with a minimum at -1 and a maximum

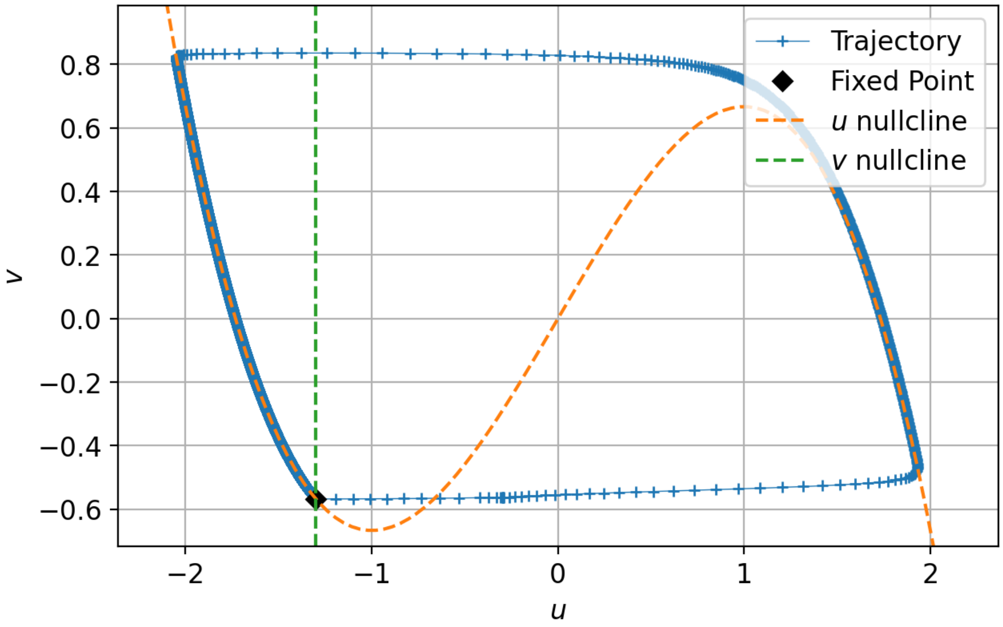

Figure 1.3.3: Phase space plot for the FHN system in the excitable state ($a = 1.3$, $\varepsilon = 0.01$). The spike is initiated via a square pulse input.

at 1. The right and leftmost branches of the nullcline are identified respectively with the physiological *refractory* and *active* phases of the neuron. In the tonic spiking regime, the limit cycle can be described as composed of four stages:

1. slow upward motion, roughly along the right (active) branch
2. fast jump towards the left branch
3. slow downward motion, roughly along the left (refractory) branch
4. fast jump towards the right branch

In the tonic activity phase, these stages are indefinitely repeated.

In the excitable phase, the fixed point becomes attractive, and the unperturbed system tends to reach it by moving along the cubic nullcline. If we consider a positive $a > 1$, for example, the fixed point will lie on the left branch of the cubic nullcline. If a quick positive shock is applied via the I_{ext} variable, the system is displaced towards the positive u's, and depending on the entity of the shock, two phenomena can take place. If the pulse is below-threshold, the system is re-captured by the refractory branch and resumes moving asymptotically towards the fixed point, whereas if the pulse is above threshold, the system jumps towards the active branch, following it up until the maximum at $u = 1$, then jumps back towards the refractory branch, resuming the asymptotic motion. This is known as a *quasi-threshold* effect, since the specific critical value of the forcing magnitude for spike generation depends also on the duration and shape of the pulse, therefore one cannot speak of a threshold in general, but only for fixed duration and shape.

By looking at the Phase Plane and performing some numerical experiments, although, we can say something more. Indeed, by evaluating the time derivative of u between the cubic nullcline branches one can find that the central branch is repulsive along the u axis. Moreover, performing simulations, and changing the pulse amplitude while keeping its duration and shape fixed, one finds that for each time that an action potential is successfully generated, the pulse has been such to displace the system *across the central nullcline branch*, although some cases in which the system crosses the nullcline, do not produce a spike. Therefore, at least approximately, the crossing of the nullcline can be considered as a threshold for action potential production.

This behaviour in the excitable state can be explained also by exploiting the explicit time-scale difference between the two variables. We proceed along the lines of the original article [Fit61]. The motion along variable u is of time scale ε, while that along variable v is of time scale 1. This allows us to consider, approximately, the motions along the two variables separately: the one along u

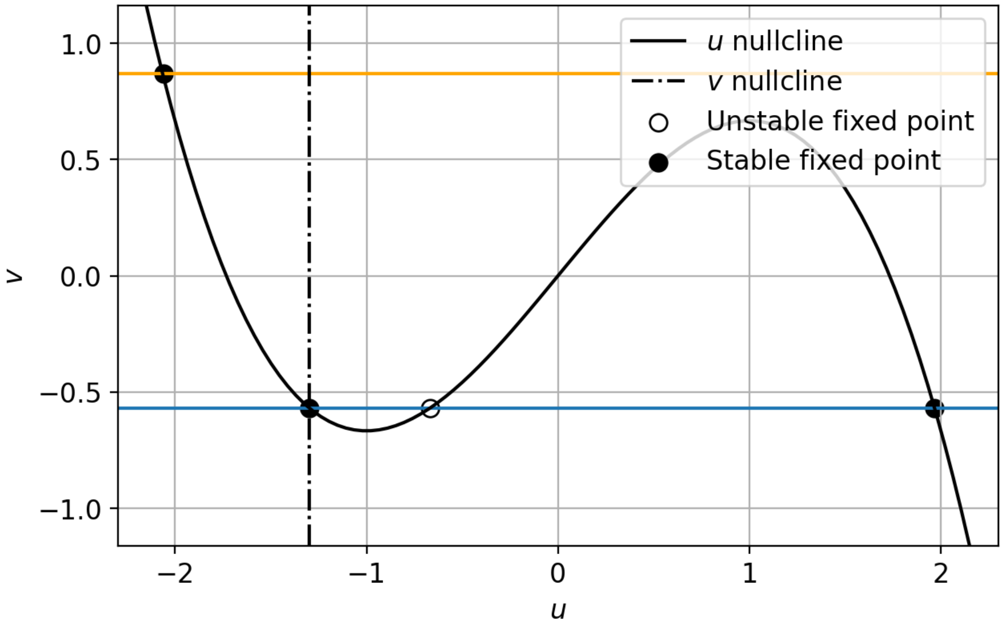

Figure 1.3.4: One-dimensional configuration space for the u variable for different values of v. Blue line: $|v| < \frac{2}{3}$, two stable and one unstable fixed point. Orange line: $|v| > \frac{2}{3}$, single stable fixed point.

can be considered to happen for constant v, the one along v can be considered to happen for a value of $u = u(v)$ that is a function of the coordinate v. Let's first consider the motion along u, which is governed by the equation

$$\varepsilon \dot{u} = u - \frac{u^3}{3} - \bar{v} \tag{1.18}$$

where $\bar{v}$ is the value of v that we consider to be fixed for evolution times of order ε. Equation (1.18) has three equilibria for $|\bar{v}| < \frac{2}{3}$, one for $|\bar{v}| > \frac{2}{3}$ and two in the limiting cases $|\bar{v}| = \pm\frac{2}{3}$. The type of equilibrium at the fixed points depends on the u-nullcline branch on which they lay, Figure 1.3.4. For $|\bar{v}| < \frac{2}{3}$, the two outermost equilibria are attractive, while the one lying on the central branch is repulsive, for $|\bar{v}| > \frac{2}{3}$, the only equilibrium is attractive, and in the limiting cases we have one attractive point and a saddle node. With these considerations in mind we can explain the production of action-potential-like spikes as follows. Consider a quiescent neuron, i.e. a FitzHugh-Nagumo system lying close to the global dynamics equilibrium point (u^*, v^*). Since we are considering the excitable regime, we can take this location to be on the left branch of the u-nullcline. The one-dimensional phase space of the ε time scale dynamics is analogous to that individuated by the blue line in Figure 1.3.4, the global fixed point corresponding to the left-most equilibrium in the reduced dynamics. Let us now force the system in the direction of positive u via a Dirac delta function pulse. Depending on the magnitude of such pulse, the system can either stop before the unstable fixed point or jump across it. In the former case the system is repelled back towards the initial equilibrium point, while in the latter it is repelled towards the stable equilibrium point on the right branch of the nullcline. Once the system has reached the right-most equibrium, the u variable remains in its proximity, while v grows, as $u + a > 0$ in the equation

$$\dot{v} = u + a \tag{1.19}$$

where since we are now considering the dynamics of time scale 1, we can take u to approximately be given by the right-branch solution of equation

$$v = u - \frac{u^3}{3} \tag{1.20}$$

therefore, as v grows, u approximately follows the right nullcline branch. As v becomes larger and larger, we cross the value $v = \frac{2}{3}$, above which the time scale ε dynamics along u suddenly remains with a single fixed point on the left branch of the u-nullcline, with a phase space analogous to that individuated by the orange line in Figure (1.3.4). In response to this change, the dynamics of order ε, rapidly moves the system towards the left branch. Once the nullcline has been reached, the motion of time scale 1 resumes, this time in a downward direction, as $u + a < 0$. Moving along the left branch of the nullcline, the system approaches the global dynamics fixed point, thus entering once again a quiescent state, awaiting stimuli. If we look at a time series, in Figure 1.3.5, we observe that the value excursion of the u variable has produced an action potential-like spike, followed by a refractory period during which it slowly tends to the fixed point. We notice as well that the v time series has maxima right after the action potential has fallen down, after which it slowly decreases back to its fixed point value, pointing out its interpretation as a recovery variable associated to the refractoriness of the neuron, which is maximal right after the action potential.

In the following we will consider only neurons in the excitable state, due to their closer resemblance to physiological neurons in the brain.

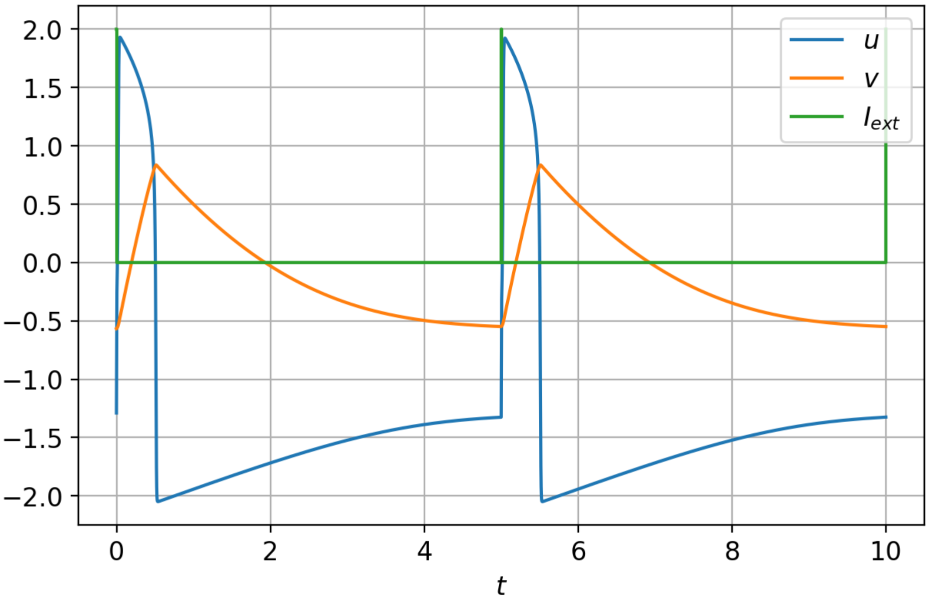

Figure 1.3.5: Time series of the dynamical variables, u in blue and v in orange, for an excitable FitzHugh-Nagumo system $\varepsilon = 0.01$, $a = 1.3$). The external stimulus, the green line, is a square wave pulse of amplitude 2, and of duration 0.001 time units, applied periodically each 10 time units.

1.4 Delay coupled Networks of model neurons

We consider a network of N FitzHugh-Nagumo neurons, coupled as follows

$$\begin{aligned} \varepsilon \dot{u}_i(t) &= u_i(t) - \frac{u_i^3(t)}{3} - v_i(t) + C \sum_{j=1}^{N} \mathcal{A}_{ij} \left[u_j\left(t-\tau\right) - u_i\left(t\right) \right] \\ \dot{v}_i(t) &= u_i(t) + a \end{aligned} \tag{1.21}$$

where $i \in \{1, \dots, N\}$ labels the single nodes, C is a coupling strength and $\mathcal{A}$ is a connectivity matrix, with $\mathcal{A}_{ij}$ representing the strength of the signal going from neuron j to neuron i. Finally, τ is a coupling delay, accounting for the finite propagation time of the action potential along the axon. The propagation time of an action potential along an axon can vary greatly, depending for example on the animal species, the type of neuron or on the axon diametre, but for medium to long range human cortical axons, it has been measured to be of order $1-5$ ms [Wan+08], and thus of the same order of the typical action potential duration, $1-5$ ms, which in our model corresponds to the slow dynamics, with characteristic time scale of order 1. This is the feature, together with the choice of the FitzHugh-Nagumo system, that makes our model a mesoscopic scale one, as we can identify the nodes with small patches of cortex, and links as connections, anatomical or functional, between them. In the following, we consider two ways to construct $\mathcal{A}$ from the adjacency matrix of the network A. We will work using the convention that $A_{ij} = 1$ represents a link going from j to i.

1. *In-degree normalised:* $\mathcal{A}_{ij} = \dfrac{A_{ij}}{\sum_j A_{ij}}$
   In this normalisation scheme, all inputs to a given node are rescaled with the inverse of the node in-degree. In this sense, the higher the in-degree of a given neuron is, the larger will the number of simultaneously firing, afferent presynaptic neurons have to be, in order for it to fire. This condition implies that $\mathcal{A}$ will be a row-stochastic matrix.

2. *Out-degree normalised:* $\mathcal{A}_{ij} = \dfrac{A_{ij}}{\sum_i A_{ij}}$
   In this scheme, each node divides equally its output signal strength among its efferent postsynaptic neurons. Even though a realistic chemical synapse model would entail a much more complicated coupling function [Rot13], we can contextualise this normalisation scheme with a subdivision, between all postsynaptic neurons, of a roughly fixed amount of neurotransmitter released by the presynaptic neuron. This condition entails a column-stochasticity condition on $\mathcal{A}$.

In the following chapter, we consider the features of synchronisation phenomena in the two aforementioned schemes. In both cases we will study synchronisation within a self-consistent framework, and we will consider for simplicity undirected architectures, but the properties of the connectivity matrix will play a relevant role in the phenomenology we will observe, and in the approach we will be able to put forward. In the following, we will consider a set of neurons to be synchronised, when its belonging neurons fire *simultaneuously*, *repeatedly* and *self-sustainedly* over time. By this we mean that, considering a synchronous set of n neurons, we will hypothesize their trajectories to be all equal to a solution $(u_s(t), v_s(t))^T$, which we call the single-neuron synchronous solution

$$\begin{pmatrix} u_1(t) \\ v_1(t) \end{pmatrix} = \begin{pmatrix} u_2(t) \\ v_2(t) \end{pmatrix} = \ldots = \begin{pmatrix} u_n(t) \\ v_n(t) \end{pmatrix} \equiv \begin{pmatrix} u_s(t) \\ v_s(t) \end{pmatrix} \tag{1.22}$$

this satisfies the requirement of *simultaneity* in the previous definition. As far as the two other requirements are concerned, they translate into conditions on $(u_s(t), v_s(t))^T$. The requisites that the neurons fire repeatedly and self-sustainedly translates into asking that $(u_s(t), v_s(t))^T$ be characterised dynamically as a limit cycle, possibly stabilised, in absence of external stimuli, by some feedback mechanism.

Chapter 2

Synchronisation phenomena in the in-degree-normalised scheme

We approach equation (1.21) by hypothesizing that, when the system is fully synchronised, all neurons follow the same trajectory.

$$u_1(t) = u_2(t) = \ldots = u_N(t) \equiv u_s(t), \quad v_1(t) = v_2(t) = \ldots = v_N(t) \equiv v_s(t) \tag{2.1}$$

By inserting these into (1.21) we obtain the following equation

$$\begin{aligned} \varepsilon \dot{u}_s(t) &= u_s(t) - \frac{u_s^3(t)}{3} - v_s(t) + C\left[u_s\left(t-\tau\right) - u_s\left(t\right)\right] \\ \dot{v}_s(t) &= u_s(t) + a \end{aligned} \tag{2.2}$$

We can see that in this scheme, the fully synchronised dynamics of the system corresponds to that of a single neuron with feedback. This is due to the stochasticity of $\mathcal{A}$, which ensures $\mathcal{A}\mathbf{1}_N = \mathbf{1}_N$. Since we have enforced the system on a synchronous trajectory by hypothesis, we must now check for the presence of *self-sustained* oscillations in the synchronous dynamics. Indeed, as a matter of principle, also a fully inactive network is completely synchronous in the sense specified by our self-consistent hypothesis, but does not meet the requirements of the definition we chose in the beginning.

2.1 Single neuron dynamical phase plot

To ensure that the synchronous solution is a self-sustained firing one, we can inspect the dynamics of the single neuron with feedback. We hypothesize that, depending on the values of τ and C, after an input spike, the system will either settle back to the fixed point or evolve towards a limit cycle dynamics. To study this hypothesis, we simulate the evolution of the system given by Equation (2.2) after a gaussian input spike, for a grid of (C, τ) values, recording N_{points} phase plane points evenly spaced of an interval of $\approx 10^{-4}$ time units one from another, for each run. We calculate the level of activity A of the neuron via

$$A(C, \tau) = \frac{\tau}{N_{points}} \sqrt{\sum_{n=1}^{N_{points}} (u_n - \bar{u})^2 + (v_n - \bar{v})^2} \tag{2.3}$$

where

$$\bar{u} = \sum_{n=1}^{N_{points}} \frac{u_n}{N_{points}}, \quad \bar{v} = \sum_{n=1}^{N_{points}} \frac{v_n}{N_{points}}$$

represent the averages of the dynamical variables for the run. Notice the τ scaling of the enumerator of A, to account for the fact that larger values of τ will lead to slower cycles, with a longer time spent in the proximity of the fixed point, awaiting feedback.

The choice of this indicator can be understood by considering the two cases we wish to distinguish, i.e. the phase point resting close to the fixed point, possibly after a small number of oscillations, and a state of self-sustained oscillations. Indeed, we are taking the time average of the modulus of displacement with respect to the mean. In the first case, apart from initial oscillations, the phase point will tend rapidly to the fixed point, and remain in its neighbourhood indefinitely, so that the mean will rapidly tend to the fixed point as well, thus yielding a value of $A(C, \tau)$ close to 0. Conversely, if we are in the self-sustained region, the phase point will be travelling along the limit cycle indefinitely, up to some resting periods near the fixed point for large τ, yielding a non-negligible modulus for the displacement with respect to the mean.

The results of the simulations are presented in Figure 2.1.1. The plot shows that for each fixed value of coupling strength, there exists a threshold value of the delay τ, which presents a decreasing trend with respect to C, and below which no self-sustained activity takes place, owing to the fact that for increasing feedback strengths, a spike can be initiated earlier and earlier along the cycle. This tendency, although, is reversed around values of $C \approx 3.5$, possibly due to the fact that for such feedback strengths, the nonlinearities in the system dynamics are unable to win over the delayed feedback term, which penalises any displacement from the fixed point. Analogously, for a fixed value of the delay term τ, there is a minimal coupling strength in order to observe self-sustained activity, which decreases, for growing τ. This can be related to the fact that, as τ grows larger and larger, the system will be resting closer and closer to the fixed point, and therefore will need smaller and smaller feedback pulses to initiate a spike.

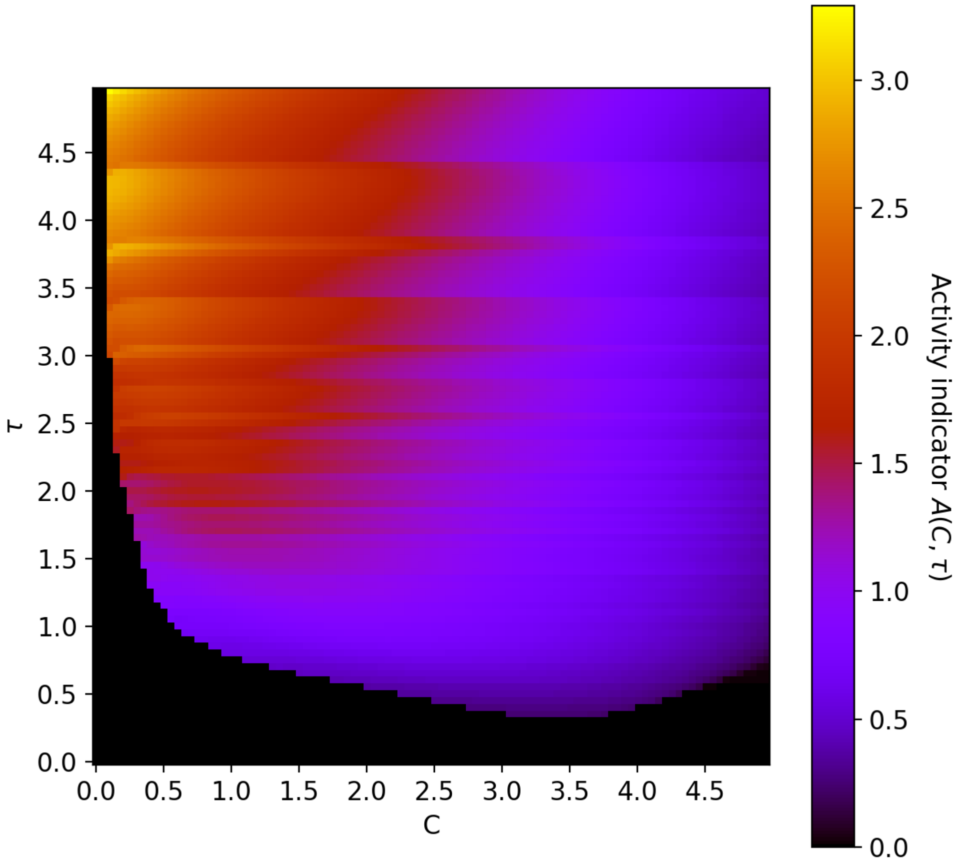

Figure 2.1.1: Plot of the activity indicator $A(C, \tau)$ evaluated along the simulated evolution of Equation (2.2), for a grid of parameter values (C, τ). We observe that the critical value of τ follows a decreasing trend with respect to C, pointing out that for larger coupling strenghts, a complete cycle can be initiated by feedback earlier and earlier along the trajectory. After $C \approx 3.5$, although, the critical τ starts growing again, possibly because in this region the feedback term overwhelms the nonlinearities, and thus keeps the system in the vicinity of the fixed point regardless of the stimulus magnitude.

These hypotheses can be corroborated by approximating the nullcline jumps as instantaneous, and therefore the feedback as a Dirac delta function, scaled by C. In this picture, the threshold C for a given τ is the minimal value for which the system is able to close the cycle in time τ. Proceeding in this way, a curve can be obtained, that qualitatively reproduces the shape of the threshold value, but gives different numerical results.

2.2 The Master Stability Function

Once we have proved the existence of the collective synchronous state via the described approach, we can set out to determine the stability of the synchronous solution. The study of the linear stability of synchronised states has been pioneered by Pecora and Carroll in 1998, with the introduction of the Master Stability Function [PC98]. The approach had been introduced at first for regular diffusive couplings without delay, but can be easily adapted to the case of delayed couplings. To this end we introduce perturbations to the synchronous dynamics of each node

$$u_i(t) = u_s(t) + \delta u_i(t) \qquad v_i(t) = v_s(t) + \delta v_i(t) \tag{2.4}$$

These perturbations evolve according to the *tangent dynamics* with respect to the synchronous solution. Let us denote by $\mathbf{e}_i$ the i-th element of the $\mathbb{R}^N$ canonical basis, and by $\mathbf{I}_N$ the identity matrix operating on the same vector space. The perturbations evolve according to

$$\begin{aligned}\sum_i \mathbf{e}_i \otimes \begin{pmatrix} \varepsilon\delta\dot{u}_i(t) \\ \delta\dot{v}_i(t) \end{pmatrix} = \mathbf{I}_N \otimes \begin{pmatrix} 1 - C - u_s^2(t) & -1 \\ 1 & 0 \end{pmatrix} \sum_i \mathbf{e}_i \otimes \begin{pmatrix} \delta u_i(t) \\ \delta v_i(t) \end{pmatrix} + \\ + (C\mathcal{A} \otimes H) \sum_i \mathbf{e}_i \otimes \begin{pmatrix} \delta u_i(t-\tau) \\ \delta v_i(t-\tau) \end{pmatrix}\end{aligned} \tag{2.5}$$

where we introduced

$$H = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \tag{2.6}$$

which is referred to as the *output function* (in this case a linear function, i.e. a matrix), that specifies how the coordinates of systems lying on adjacent nodes are coupled. Now let $B^{-1}\mathcal{A}B = S$, with $S = \mathrm{diag}(\sigma_1, \dots, \sigma_N)$. In our case, the underlying network is undirected A is symmetrical and as such has a real spectrum. Even though our normalisation choice breaks the explicit symmetry of A, as we pass to $\mathcal{A}$, the spectrum remains real [1]. Were we to consider directed

[1] Matrices such as $\mathcal{A}$ are often referred to as symmetrizable, as they can be cast to a symmetric form via multiplication by a diagonal matrix. In our case $A = D\mathcal{A}$.

networks, and therefore "genuinely" non-symmetric matrices, we would need to extend the domain of the eigenvalues to the complex plane. We can multiply (2.5) on the left by $B^{-1} \otimes \mathbf{I}_2$, and by inserting $(B^{-1} \otimes \mathbf{I}_2)\,(B \otimes \mathbf{I}_2)$ pairs in the r.h.s. we get

$$
\begin{aligned}
\sum_i B^{-1}\mathbf{e}_i \otimes \begin{pmatrix} \varepsilon\delta\dot{u}_i(t) \\ \delta\dot{v}_i(t) \end{pmatrix} &= \\
&= \left(B^{-1}B \otimes \begin{pmatrix} 1 - C - u_s^2(t) & -1 \\ 1 & 0 \end{pmatrix} \right) \sum_i B^{-1}\mathbf{e}_i \otimes \begin{pmatrix} \delta u_i(t) \\ \delta v_i(t) \end{pmatrix} + \\
&\quad + \left(CB^{-1}\mathcal{A}B \otimes H \right) \sum_i B^{-1}\mathbf{e}_i \otimes \begin{pmatrix} \delta u_i(t-\tau) \\ \delta v_i(t-\tau) \end{pmatrix} \qquad (2.7)
\end{aligned}
$$

Denoting $\boldsymbol{\eta}_i = B^{-1}\mathbf{e}_i$ the new base vectors, we can write

$$
\begin{aligned}
\sum_i \boldsymbol{\eta}_i \otimes \begin{pmatrix} \varepsilon\delta\dot{u}_i(t) \\ \delta\dot{v}_i(t) \end{pmatrix} &= \left(\mathbf{I}_N \otimes \begin{pmatrix} 1 - C - u_s^2(t) & -1 \\ 1 & 0 \end{pmatrix} \right) \sum_i \boldsymbol{\eta}_i \otimes \begin{pmatrix} \delta u_i(t) \\ \delta v_i(t) \end{pmatrix} + \\
&\quad + (CS \otimes H) \sum_i \boldsymbol{\eta}_i \otimes \begin{pmatrix} \delta u_i(t-\tau) \\ \delta v_i(t-\tau) \end{pmatrix} \qquad (2.8)
\end{aligned}
$$

We observe that, due to the diagonal form of S we have effectively decoupled the perturbations. In this sense we can drop the $\boldsymbol{\eta}_i$ and write the equation for each mode as

$$
\begin{pmatrix} \varepsilon\delta\dot{u}_i(t) \\ \delta\dot{v}_i(t) \end{pmatrix} = \begin{pmatrix} 1 - C - u_s^2(t) & -1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} \delta u_i(t) \\ \delta v_i(t) \end{pmatrix} + \sigma_i CH \begin{pmatrix} \delta u_i(t-\tau) \\ \delta v_i(t-\tau) \end{pmatrix} \qquad (2.9)
$$

Notice that now i does not index localised perturbations any more, but rather delocalised perturbation over the whole network. If we gather C and σ_i into a single parameter ν, we obtain Equation (2.10), known as the *Master Stability Equation*, which allows to simplify the study of the stability of a synchronous solution.

$$
\begin{pmatrix} \varepsilon\delta\dot{u}(t) \\ \delta\dot{v}(t) \end{pmatrix} = \begin{pmatrix} 1 - C - u_s^2(t) & -1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} \delta u(t) \\ \delta v(t) \end{pmatrix} + \nu H \begin{pmatrix} \delta u(t-\tau) \\ \delta v(t-\tau) \end{pmatrix} \qquad (2.10)
$$

In the tangent dynamics, perfect synchronisation corresponds to the zero solution. Therefore, we assess the stability of synchronisation, by evaluating numerically the *maximal Lyapunov exponent* for the zero solution as a function of $\nu = \sigma C$. This function is called the *Master Stability Function* $\Lambda(\nu)$. A plot of the MSF for the completely synchronous solution of the FitzHugh-Nagumo system is in Figure 2.2.1. The numerical calculation has been simplified by noticing that since the synchronous solution (u_s, v_s) is periodic, with a period $T_s \approx \tau$, the solution to the Master Stability Equation can be searched for in a Floquet form

$$\begin{pmatrix} \delta u(t) \\ \delta v(t) \end{pmatrix} = e^{(\lambda + i\omega)t} \begin{pmatrix} \delta \tilde{u}(t) \\ \delta \tilde{v}(t) \end{pmatrix} \tag{2.11}$$

where

$$\begin{pmatrix} \delta \tilde{u}(t) \\ \delta \tilde{v}(t) \end{pmatrix} = \begin{pmatrix} \delta \tilde{u}(t + k\tau) \\ \delta \tilde{v}(t + k\tau) \end{pmatrix} \quad k \in \mathbb{Z} \tag{2.12}$$

is periodic with period τ. It can be proved [see e.g. Tur98] that the λ's correspond to the Lyapunov exponents for the Poincaré map, i.e. the map of the solution over a period τ. This greatly lightens the simulations as it becomes sufficient to find the evolution of an initial perturbation for multiples of τ, i.e. to solve numerically for the Poincaré map, as it is the value of λ to determine the shrinking or expanding properties of the solution to the Master Stability Equation. We adopted this strategy, and obtained the values of Λ shown in the plots by performing a linear regression in logarithmic scale over the trajectories of the Poincaré map. The stability of the synchronous solution with respect to perturbations is confirmed if all rescaled eigenvalues of the connectivity matrix fall within the negative domain of $\Lambda(\nu)$.

For the FitzHugh-Nagumo system, previous studies [Leh16], [Leh+11] have derived a constraint for the boundary of the negative domain of $\Lambda(\nu)$. The studies consider in principle a complex spectrum, and treat the stability problem for the synchronous solution along the lines of Pyragas delayed feedback control for chaotic systems. The authors concern themselves with the problem of determining the boundary of the negative domain, i.e. the curve on which the Master Stability Function is zero. The constraint on the critical eigenvalue ν_c is derived in the form

$$\mathrm{Re}(\nu_c)^2 + \mathrm{Im}(\nu_c)^2 = C^2 + K \tag{2.13}$$

where ν_c is the critical rescaled eigenvalue, for which $\Lambda(\nu_c) = 0$, and K is a positive constant, function of some dynamical properties of the synchronous solution that in general cannot be calculated. Even without knowing the form of K, the previous formula becomes useful as a lower bound, since whenever a scaled eigenvalue falls within the circle of center $\nu = 0$ and radius C, the Master Stability Function is guaranteed to be negative, the sole exceptions are allowed to happen on the boundary, where the Master Stability Function can be negative.

In our specific case, we are considering a real spectrum matrix $\mathcal{A}$, therefore the lower bound is restricted to the real interval $[-C, C]$. Moreover, we observe

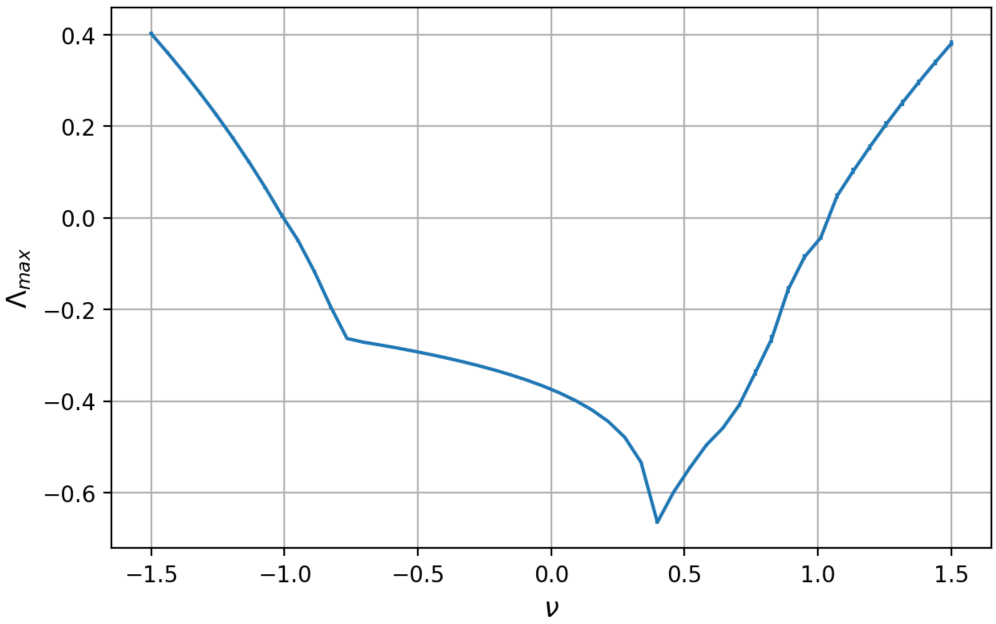

Figure 2.2.1: The Master Stability Function for the FitzHugh-Nagumo system. The delay has been set to $\tau = 1$, the time-scale separation to $\varepsilon = 0.01$.

that in this coupling scheme, apart from the conditions for the existence of a self-sustained solution in the first place, the coupling constant C does not play a role in the stability analysis, since by a simple scaling we can rephrase the bound on the eigenvalues independently of it

$$\nu = \sigma C \in [-C, C] \leftrightarrow \sigma \in [-1, 1] \tag{2.14}$$

Since for a stochastic matrix the spectrum is bound to lie in $[-1, 1]$, and since $\mathcal{A}$ is stochastic, we conclude that the only possible eigenspaces along which non vanishing perturbations to the synchronous state can arise, are those associated to the eigenvalues -1, and 1 corresponding to the boundary of the negative region.

By construction, the eigenspace associated to 1 is spanned by $\mathbf{1}_N$, and is therefore associated to perturbations along the completely synchronous dynamics. Therefore, the only perturbations *transversal* to complete synchrony, are those along the -1 eigenspace. With this analysis, therefore, we have ascertained that if the connectivity matrix is row-stochastic, the completely synchronous solution is always stable, except for perturbations along the eigenspace associated to eigenvalue -1. In the following section, we will see under which conditions this eigenspace exists, and how we can introduce it by modifying iteratively the network.

2.3 Destabilising synchrony

Since we are considering an undirected network, we can write $d_i = \sum_{j=1}^{N} A_{ij}$, as the in and out degree of each node coincide. We can also build $D = \text{diag}\{d_1, \ldots, d_N\}$, the so called degree matrix. With this notation introduced, $\mathcal{A}$ can be written as $\mathcal{A} = D^{-1}A$. One notices immediately that defining $L_{RW} = I - D^{-1}A$, the random-walk-normalised Laplacian, one has $[\mathcal{A}, L_{RW}] = 0$, and denoting the sorted spectrum of $\mathcal{A}$ as $\{\sigma_i\}_{1\leq i\leq N}$, and that of L_{RW} as $\{\lambda_i\}_{1\leq i\leq N}$, one has

$$\sigma_i = 1 - \lambda_{N-i+1} \tag{2.15}$$

from which we deduce that $\sigma_1 = -1 \iff \lambda_N = 2$. As we stated previously, in Theorem 1.2.2, $\lambda_N = 2$ if and only if the graph is bipartite.

This fact, together with the results from the previous section, guarantees that the only networks on which a neutrally stable perturbation transversal to complete synchrony can arise, are bipartite.

Let us consider a bipartite graph, composed of partitions P and Q. The bipartition condition implies that there exist $\mathbf{1}_P$ and $\mathbf{1}_Q$, such that

$$\begin{gathered} \mathcal{A}\mathbf{1}_P = \mathbf{1}_Q \qquad \mathcal{A}\mathbf{1}_Q = \mathbf{1}_P \qquad \mathbf{1}_P + \mathbf{1}_Q = \mathbf{1}_N \\ \mathbf{1}_P = \frac{1}{2}\left(\mathbf{1}_N + \mathbf{a}_1\right) \qquad \mathbf{1}_Q = \frac{1}{2}\left(\mathbf{1}_N - \mathbf{a}_1\right) \end{gathered} \tag{2.16}$$

where by $\mathcal{A}$ we denote the unit-row sum connectivity matrix corresponding to the bipartite structure, and with $\mathbf{a}_1$ its eigenvector associated to the smallest eigenvalue, which in this case is -1. It can be seen in simulations that bipartite architectures can support bipartite solutions, i.e. solutions where two partitions fire in an alternating fashion. We can search for them by adapting the self-consistent strategy. Up to a permutation of node labels these solutions can be written as

$$\begin{aligned} u_1(t) = \ldots = u_n(t) = u_P(t) \qquad u_{n+1}(t) = \ldots = u_N(t) = u_Q(t) \\ v_1(t) = \ldots = v_n(t) = u_P(t) \qquad v_{n+1}(t) = \ldots = v_N(t) = v_Q(t) \end{aligned} \tag{2.17}$$

if we take P and Q to contain n and $N - n$ nodes each. Inserting these conditions in the dynamics we obtain

$$\begin{aligned}
&\mathbf{1}_P \otimes \begin{pmatrix} \varepsilon \dot{u}_P(t) \\ \dot{v}_P(t) \end{pmatrix} + \mathbf{1}_Q \otimes \begin{pmatrix} \varepsilon \dot{u}_Q(t) \\ \dot{v}_Q(t) \end{pmatrix} = \\
&= \mathbf{1}_P \otimes \begin{pmatrix} u_P(t) - \dfrac{u_P^3(t)}{3} - v_P(t) \\ u_P(t) + a \end{pmatrix} + \mathbf{1}_Q \otimes \begin{pmatrix} u_Q(t) - \dfrac{u_Q^3(t)}{3} - v_Q(t) \\ u_Q(t) + a \end{pmatrix} + \\
&\qquad + C\left(\mathcal{A} \otimes H\right)\left(\mathbf{1}_P \otimes \begin{pmatrix} u_P(t-\tau) \\ v_P(t-\tau) \end{pmatrix} + \mathbf{1}_Q \otimes \begin{pmatrix} u_Q(t-\tau) \\ v_Q(t-\tau) \end{pmatrix}\right) + \\
&\qquad\qquad - C\left(\mathbf{I}_N \otimes H\right)\left(\mathbf{1}_P \otimes \begin{pmatrix} u_P(t) \\ v_P(t) \end{pmatrix} + \mathbf{1}_Q \otimes \begin{pmatrix} u_Q(t) \\ v_Q(t) \end{pmatrix}\right)
\end{aligned} \tag{2.18}$$

By using (2.16) we can remove $\mathcal{A}$ from the expression

$$\begin{aligned}
&\mathbf{1}_P \otimes \begin{pmatrix} \varepsilon \dot{u}_P(t) \\ \dot{v}_P(t) \end{pmatrix} + \mathbf{1}_Q \otimes \begin{pmatrix} \varepsilon \dot{u}_Q(t) \\ \dot{v}_Q(t) \end{pmatrix} = \\
&= \mathbf{1}_P \otimes \begin{pmatrix} u_P(t) - \dfrac{u_P^3(t)}{3} - v_P(t) \\ u_P(t) + a \end{pmatrix} + \mathbf{1}_Q \otimes \begin{pmatrix} u_Q(t) - \dfrac{u_Q^3(t)}{3} - v_Q(t) \\ u_Q(t) + a \end{pmatrix} + \\
&+ C\left(\mathbf{I}_N \otimes H\right)\left(\mathbf{1}_Q \otimes \begin{pmatrix} u_P(t-\tau) - u_Q(t) \\ v_P(t-\tau) - v_Q(t) \end{pmatrix} + \mathbf{1}_P \otimes \begin{pmatrix} u_Q(t-\tau) - u_P(t) \\ v_Q(t-\tau) - v_P(t) \end{pmatrix}\right)
\end{aligned} \tag{2.19}$$

and therefore, by using the structural properties of the network, we are again able to reduce the dimensionality of the problem, this time to a 2 × 2 dimensional problem, where each system acts as a forcing term upon the other.

$$\begin{aligned}
\varepsilon \dot{u}_P(t) &= u_P(t) - \frac{u_P^3(t)}{3} - v_P(t) + C\left(u_Q(t-\tau) - u_P(t)\right) \\
\dot{v}_P(t) &= u_P(t) + a \\
\varepsilon \dot{u}_Q(t) &= u_Q(t) - \frac{u_Q^3(t)}{3} - v_Q(t) + C\left(u_P(t-\tau) - u_Q(t)\right) \\
\dot{v}_Q(t) &= u_Q(t) + a
\end{aligned} \tag{2.20}$$

The stability analysis of a bipartite synchronous state can be performed as well in the Master Stability Function framework. The adaptation of the Master Stability Function approach to multipartite synchrony is presented in [Leh16], to which the interested reader is invited to refer for full detail, and it exploits the off-diagonal block structure of multipartite architectures to find a basis change that can cast the evolution equation for the perturbations in a diagonal form,

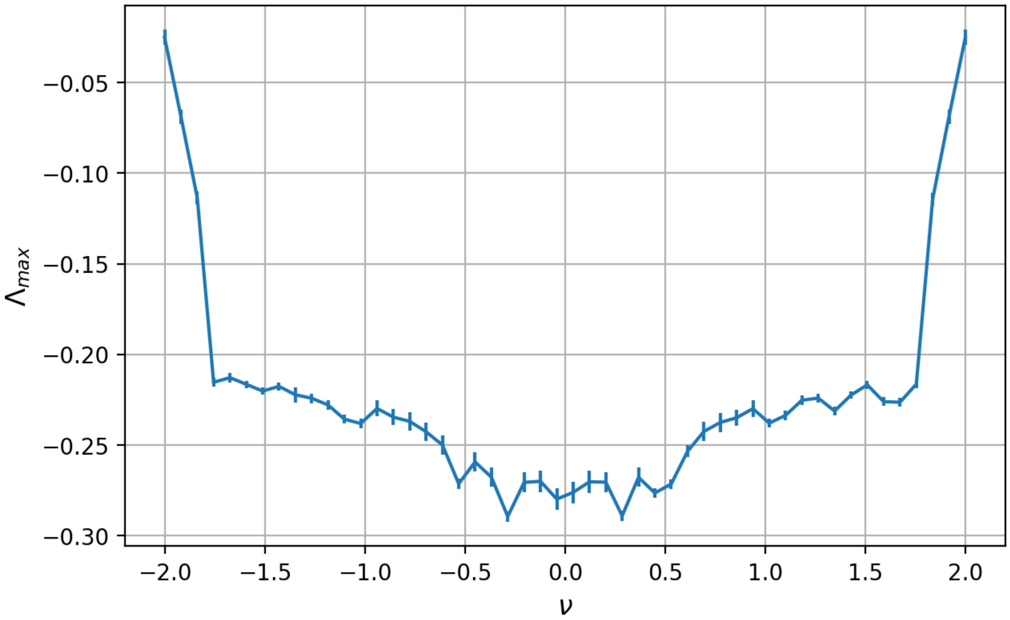

Figure 2.3.1: Master Stability Function for the bipartite solution. The parameters used in the simulation were $\varepsilon = 0.01$, $a = 1.3$, $\tau = 1$, $C = 1$.

so to obtain an analogous result to what we have shown for the simpler case of complete synchronisation. The Master Stability Equation for the bipartite stable state is 2×2 dimensional, and reads

$$\begin{aligned} \varepsilon \delta \dot{u}_P(t) &= \left(1 - u_P^2(t)\right) \delta u_P(t) - \delta v_P(t) + \nu \left(\delta u_Q(t-\tau) - \delta u_P(t)\right) \\ \delta \dot{v}_P(t) &= \delta u_P(t) \\ \varepsilon \delta \dot{u}_Q(t) &= \left(1 - u_Q^2(t)\right) \delta u_Q(t) - \delta v_Q(t) + \nu \left(\delta u_P(t-\tau) - \delta u_Q(t)\right) \\ \delta \dot{v}_Q(t) &= \delta u_Q(t) \end{aligned} \tag{2.21}$$

As in the previous case $\nu = C\sigma_i$ takes in principle the value of the scaled eigenvalues of $\mathcal{A}$, but can be used to study the equation parametrically, so to build a Master Stability Function. We proceed in analogy to the previous case, and study the maximum Lyapunov exponent of the zero solution of Equation (2.21), thus building another Master Stability Function, $\Lambda_B(\nu)$ for the stability of bipartite synchrony. The resulting function, obtained via numerical simulation, with analogous methods to those employed for complete synchronisation, is presented in Figure 2.3.1.

The simulation shows clearly that the Lyapunov exponent is negative within the whole interval $[-1,1]$, within which the spectrum of the coupling matrix must fall, on grounds of stochasticity. We can therefore state that the bipartite state is always attractive on a bipartite network, within this normalisation scheme. With these results in mind, it appears clear that within this scheme, the only way to destabilise complete synchrony, without fully removing neural activity, is to reroute the connections in such a way to move the architecture towards a bipartite network.

To this end, we develop an algorithm, based on a slight adaptation of a Spectral Centrality measure.

2.4 Spectral Centrality

In [PR12], the measure of k-spectral centrality is introduced. It aims to quantify the impact that the removal of a graph subset $\mathcal{B}$ has on the k-th smallest Laplacian eigenvalue. This is proposed as a centrality measure, due to the relevant properties of the Laplacian in diffusion processes on networks, and to the rich algebraic characterisation of the network that can be performed via its spectrum. Let us denote A the adjacency matrix before removal, A' the adjacency matrix after the removal of $\mathcal{B}$, L the Laplacian before the removal of the set under study, and L' the Laplacian after removal. We also set $\Delta A = A' - A$, $\Delta L = L' - L$.

Consider the network before removal, with adjacency matrix A, being deformed continuously on the set $\mathcal{B}$, with the strength of the deformation being controlled by the continuous parameter ε. Its Laplacian reads

$$L(\varepsilon) = \varepsilon L' + (1-\varepsilon)L = L + \varepsilon \Delta L \qquad \varepsilon \in [0,1] \tag{2.22}$$

Let $\lambda_k\left(\varepsilon\right)$ represent the k-th smallest Laplacian eigenvalue. The k-th spectral centrality of the set $\mathcal{B}$ is defined as

$$s_{\mathcal{B}}^k = \left|\frac{\partial \lambda_k(\varepsilon)}{\partial \varepsilon}\right|_{\varepsilon=0} \tag{2.23}$$

We show that we can link this expression to a perturbative correction to lowest order. First we state a known fact about the derivative of an eigenvalue w.r.t. a parameter [see e.g. Lax07, p. 134]. Consider a square matrix function $M\left(\varepsilon\right)$, differentiable in ε. Let it admit an eigenvalue $m(\varepsilon)$ associated to the right eigenvector $\vec{u}(\varepsilon)$ and to the left eigenvector $\vec{v}(\varepsilon)$. Excluding cases of eigenvalue coalescence or splitting, the derivative of m with respect to the parameter ε is given by

$$\dot{m}(\varepsilon) = \frac{\vec{v}(\varepsilon) \cdot \dot{M}(\varepsilon)\vec{u}(\varepsilon)}{\vec{v}(\varepsilon) \cdot \vec{u}(\varepsilon)} \tag{2.24}$$

where we have denoted by the dot the derivative with respect to the parameter. The k-Spectral Centrality being defined as $|\dot{\lambda}_k(0)|$, we can apply the previous formula to the Laplacian to calculate it

$$\begin{aligned} \dot{\lambda}(0) &= \frac{\vec{v}_k(0) \cdot \dot{L}(0)\vec{v}_k(0)}{\vec{v}_k(0) \cdot \vec{v}_k(0)} \\ &= \frac{1}{|\vec{v}_k|^2}\vec{v}_k \cdot \frac{\partial}{\partial \varepsilon}\left(\varepsilon L' + (1-\varepsilon)L\right)\big|_{\varepsilon=0}\vec{v}_k \\ &= \frac{\vec{v}_k \cdot (L' - L)\,\vec{v}_k}{|\vec{v}_k|^2} \equiv \frac{\vec{v}_k \cdot \Delta L \vec{v}_k}{|\vec{v}_k|^2} \end{aligned} \tag{2.25}$$

where, in the second and third line, we call for brevity $\vec{v}_k(0) = \vec{v}_k$. By taking the absolute value one gets $s^k_{\mathcal{B}}$. In this sense, therefore, we can write the expression for Spectral Centrality as we would write the lowest order perturbative correction given by the operator ΔL, corresponding to the finite change underwent by the Laplacian because of the removal of the graph subset $\mathcal{B}$ from the network.

2.5 Constrained Spectral Centrality

We want to use the notion of Spectral Centrality, albeit without taking the absolute value, in order to algorithmically select which links to cut in the network, so to move the structure the closest to a bipartite graph, possibly reaching exactly one.

Therefore, we have to modify slightly the previous definition. Indeed, in order to remain within the same normalisation scheme as the network is modified, we must take into account only network modifications that preserve the unit-row-sum condition on $\mathcal{A}$, and work with the random-walk normalised Laplacian.

Consider an undirected weighted graph of N nodes without self-loops, with adjacency structure specified by the $N \times N$ matrix A. Let us consider separately the matrix W of weights associated to each edge, $W_{ij} = \frac{1}{d_i}$ if $A_{ij} = 1$, which fullfills a unit-row-sum constraint $\sum_j W_{ij} = 1$ by construction. The normalised Laplacian can be written directly as

$$L_{RW} = I - W \circ A = I - D^{-1}A = I - \mathcal{A} \tag{2.26}$$

with $\circ$ denoting the Hadamard product. The separation of $\mathcal{A} = W \circ A$ has been made for ease of use in the calculations.

We consider the effect of the removal of edge (l, m) from the graph, on the k-th eigenvalue. Since we want to preserve the unit-row-sum condition of $\mathcal{A}$, we need to choose a *weight rerouting policy* to apply when cutting an edge. In the following we will consider an even redistribution among the remaining links. This means that after cutting link (l, m) we set $A_{lm} = A_{ml} = 0$ in the adjacency matrix and $W_{lm} = W_{ml} = 0$ in the weight matrix, but we also add a contribution $\frac{W_{lm}}{d_l - 1}$ to all remaining nonzero entries in row l of W, and analogously a contribution $\frac{W_{ml}}{d_m - 1}$ to the nonzero entries in row m, where by d_i we mean the degree of node i *before* the cut.
Therefore, the matrix elements *after* the cut and the weight rerouting read

$$A'_{ij} = A_{ij} \left(1 - \delta_{il}\delta_{jm} - \delta_{im}\delta_{jl}\right) \tag{2.27}$$

$$W'_{ij} = W_{ij} \left(1 - \delta_{il} - \delta_{im}\right) + \delta_{il} \left(1 - \delta_{jm}\right) \left(W_{ij} + \frac{W_{im}}{d_i - 1}\right) + \tag{2.28}$$

$$+ \delta_{im} \left(1 - \delta_{jl}\right) \left(W_{ij} + \frac{W_{il}}{d_i - 1}\right) \tag{2.29}$$

To calculate the constrained k-spectral centrality, we can just use the formula in (2.25). We just have to consider the normalised Laplacian L_{RW} instead of the regular one, and notice that the variation in L_{RW} amounts to $\Delta L_{RW} = -\Delta\mathcal{A}$, if we take care to re-normalise the weights after the cut. First we calculate $\mathcal{A}'$, *after* the cut

$$\begin{aligned}\mathcal{A}'_{ij} &= \left[A' \circ W'\right]_{ij} = \\ &= A_{ij}W_{ij} \left(1 - \delta_{il} - \delta_{im}\right) + A_{ij} \left(W_{ij} + \frac{W_{im}}{d_i - 1}\right) \left(\delta_{il} - \delta_{il}\delta_{jm}\right) \\ &+ A_{ij} \left(W_{ij} + \frac{W_{il}}{d_i - 1}\right) \left(\delta_{im} - \delta_{im}\delta_{jl}\right)\end{aligned} \tag{2.30}$$

By subtracting $\mathcal{A}$ we readily obtain the finite variation of the connectivity matrix, which is equal to minus the variation of the normalised Laplacian.

$$\Delta\mathcal{A} = \mathcal{A}' - \mathcal{A} = -\Delta L_{RW} \tag{2.31}$$

$$\begin{aligned}\Delta\mathcal{A}_{ij} = &- \left(\delta_{il} + \delta_{im}\right) A_{ij}W_{ij} + A_{ij} \left(W_{ij} + \frac{W_{im}}{d_i - 1}\right) \left(\delta_{il} - \delta_{il}\delta_{jm}\right) \\ &+ A_{ij} \left(W_{ij} + \frac{W_{il}}{d_i - 1}\right) \left(\delta_{im} - \delta_{im}\delta_{jl}\right)\end{aligned} \tag{2.32}$$

We have seen that the expression for k-spectral centrality can be led back to the perturbative expression for the change to the k-th eigenvalue

$$\dot{\lambda}_k\left(0\right) = \Delta\lambda_k = \frac{\vec{v}_k \cdot \Delta L \vec{v}_k}{|\vec{v}_k|^2} \tag{2.33}$$

We set out to evaluate the adapted version of quantity, dropping the index k in the following for ease of notation, and understanding that $\vec{v}$, $\vec{u}$ and λ are respectively the left eigenvector, the right eigenvector and their common eigenvalue[2].

$$\begin{aligned}\Delta\lambda &= -\frac{\vec{v}\cdot\mathcal{A}\vec{u}}{\vec{v}\cdot\vec{u}} = & (2.34)\\ &= \frac{A_{lm}}{\vec{v}\cdot\vec{u}}\left[\frac{d_l}{d_l-1}W_{lm}v_l u_m + \frac{d_m}{d_m-1}W_{ml}v_m u_l\right] + & (2.35)\\ &-\sum_j\left[\frac{v_l W_{lm}}{d_l-1}A_{lj} + \frac{v_m W_{ml}}{d_m-1}A_{mj}\right]\frac{u_j}{\vec{v}\cdot\vec{u}} & (2.36)\end{aligned}$$

Now, substituting $W_{lm} = \frac{1}{d_l}$ and $W_{ml} = \frac{1}{d_m}$ we obtain the expression for the signed spectral centrality of edge (l, m).

$$\dot{\lambda}(0) = \frac{1}{\vec{v}\cdot\vec{u}}\left[\frac{\left(\lambda-1\right)v_l u_l + v_l u_m}{d_l-1} + \frac{\left(\lambda-1\right)v_m u_m + v_m u_l}{d_m-1}\right] \tag{2.37}$$

an analogous expression to the one found in [He+19], in a slighty different context. This expression allows us to estimate, to lowest perturbative order, the impact that removal of link (l, m), with a subsequent re-normalisation of coupling strenghts, will have on the eigenvalue λ of the normalised Laplacian spectrum.

2.6 Bipartitioning algorithm

The aim of our algorithm is to select and cut links progressively from a graph, in order to obtain a bipartite network. In doing so, the weights are renormalised so to maintain the unit-row-sum condition on $\mathcal{A}$. An important caveat in this operation, is to avoid disconnecting the graph. The algorithm proceeds iteratively. The evaluation of the most favourable links to cut is performed by using the signed spectral centrality expression in (2.37). The expression is used to

[2] The random-walk normalisation breaks the explicit symmetry of the Laplacian, and therefore one must account for different left and right eigenvectors, even if each right eigenvector differs from its left counterpart only by a multiplication by D, the degree matrix, and the spectrum remains real.

select a list of links which would contribute by moving λ_N towards 2. This list is then sorted, and starting from the largest contributing edge, the graph is checked for connectedness upon removal of said edge. If the graph remains connected, the removal is accepted, and a new iteration of perturbative calculations begins with the updated graph. Otherwise, the move is rejected, and the candidate with the next largest contribution is considered. These steps are repeated until λ_N is closer than a given tolerance *tol*, initialised by default to 10^{-6}, to the value 2. This tolerance has been introduced to account for finite precision effects, but in general we observed that it was not necessary, as once the network is bipartite, $\lambda_N = 2$ is correct to a very high precision. Errors are raised if no link gives a positive contribution to λ_N or if no candidate link can be removed without disconnecting the graph. A pseudocode describing the algorithm in detail is given in Algorithm 1, in Appendix C.

The algorithm has been implemented in Python, using the `networkx` module [HSS08] in order to handle networks. In particular, the connectedness check has been implemented via the `networkx.is_connected` function, which implements an optimised combinatorial algorithm, since during development, the usage of an algebraic indicator, such as the Fiedler value, has proved hard to balance correctly within a cost function.

2.7 Algorithm testing

To test the algorithm we generate networks according to three different random models, and verify that within the chosen normalisation scheme, they all synchronise. Then we apply the bipartition algorithm, and check that, for the same configuration of initial stimuli, the network evolves instead towards a bipartite stable state. The initial impulse is administered in one every four neurons. The impulse has the shape of a sharp Gaussian, with $\sigma = 0.025$ and amplitude $A = 2$. The dynamical parameters used in all simulations are $\tau = 1$, $\varepsilon = 0.01$, $a = 1.3$ and $C = 0.6125$. The value has been chosen to be close to $C^*(1) \approx 0.6125$, the critical point for $\tau = 1$. The simulations have been done using Python driver programs. All the numerical integrations have been performed using the `RADAR5` algorithm [GH05], about which some details be found in Appendix C, via its Python interface `pyradar5`, and the network handling has been performed using `networkx`. Due to the computational cost of simulation, the size of the networks has been limited to 50 nodes in this phase.

Erdős-Renyi network An undirected Erdős-Renyi network [ER59] with 50 nodes has been generated, with linking probability $p = 0.25$, ending up with

301 edges. The simulations respectively before and after applying the algorithm are presented in Figures 2.7.1 and 2.7.2. The Figures show the time series for the u variable for each of the neurons in the network. In Figure 2.7.1 the system evolves immediately towards complete synchrony, whereas in Figure 2.37 the sytem evolves towards a bipartite stable state.

Watts-Strogatz network An undirected Watts-Strogatz network [WS98] with 50 nodes has been generated, with $k = 4$ nearest neighbours per node, thus totalling 100 edges, and rewiring probability $p = 0.25$. The simulations respectively before and after applying the algorithm are presented in Figures 2.7.3 and 2.7.4. In Figure 2.7.3 we see that the onset of complete synchrony is somewhat slower than in the Erdős-Renyi case. After applying the algorithm the system settles towards a bipartite stable state, but also in this case the phenomenon is slower than what we saw in the E-R case, with the remains of some completely synchronous activity still appearing as under-threshold oscillations in some neurons, in the last 10 time units in Figure 2.7.4.

Barabási-Albert network An undirected Barabasi-Albert network [BA99] with 50 nodes has been generated, with attachment parameter $m = 4$. The simulations respectively before and after applying the algorithm are presented in Figures 2.7.5 and 2.7.6. In Figure 2.7.5 we see that the network quickly settles towards a completely synchronous state. After applying the algorithm the system tends towards bipartite synchrony, but similarly to what we observed in the Watts-Strogatz case, there are some transient remains of complete synchrony, in the trace of some neurons.

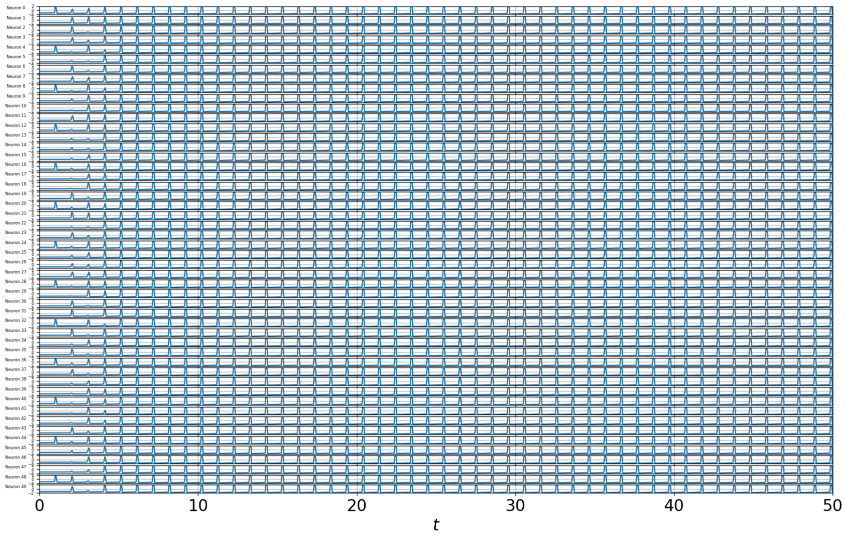

Figure 2.7.1: Simulation on the Erdős-Renyi network, before applying the Bipartition Algorithm: the system evolves towards complete synchrony.

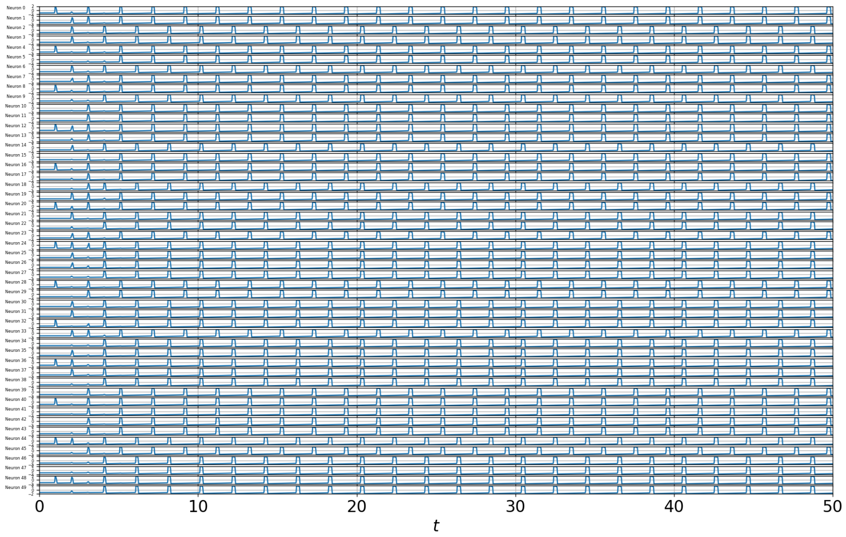

Figure 2.7.2: Simulation on the Erdős-Renyi network, after applying the Bipartition Algorithm: the system evolves towards bipartite synchrony.

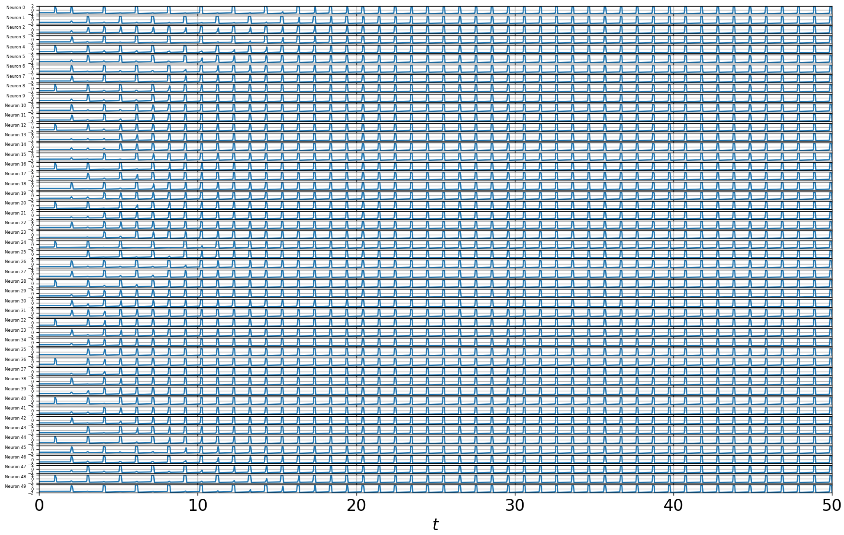

Figure 2.7.3: Simulation on the Watts-Strogatz network, before applying the Bipartition Algorithm: the system evolves towards complete synchrony.

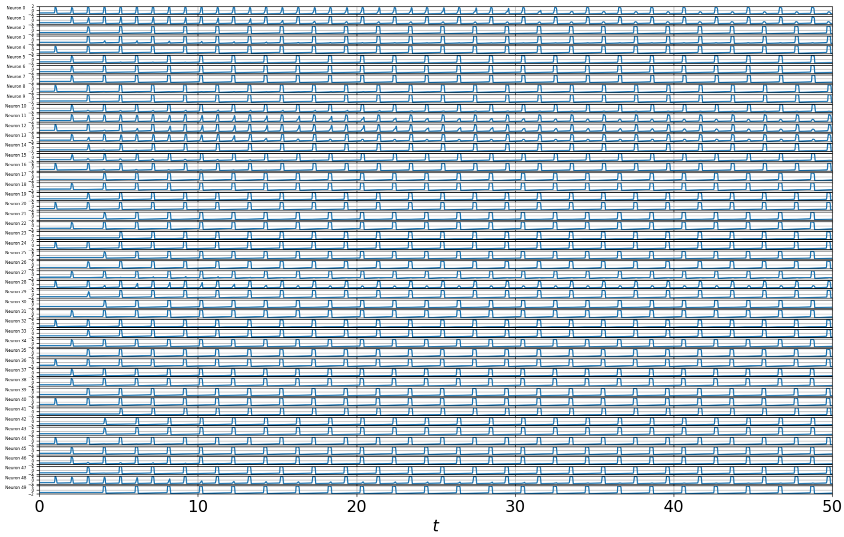

Figure 2.7.4: Simulation on the Watts-Strogatz network, after applying the Bipartition Algorithm: the system evolves towards bipartite synchrony.

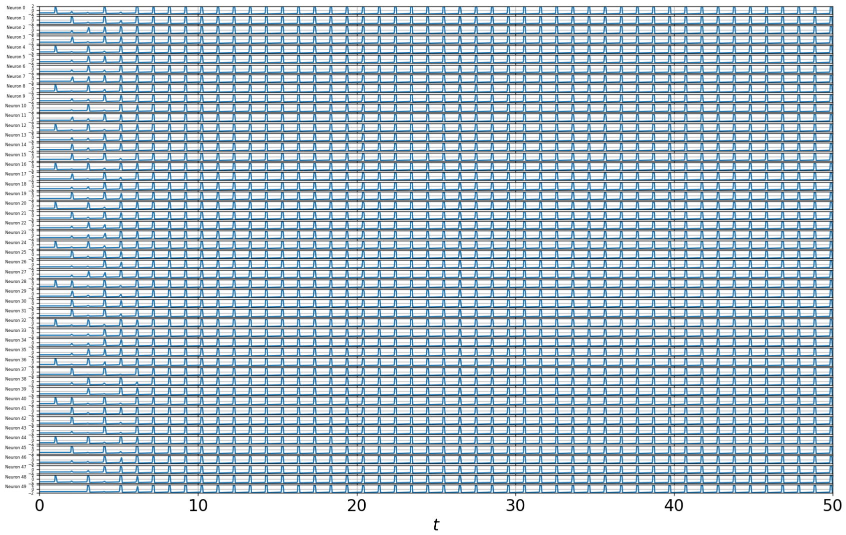

Figure 2.7.5: Simulation on the Barabasi-Albert network, before applying the Bipartition Algorithm: the system evolves towards complete synchrony.

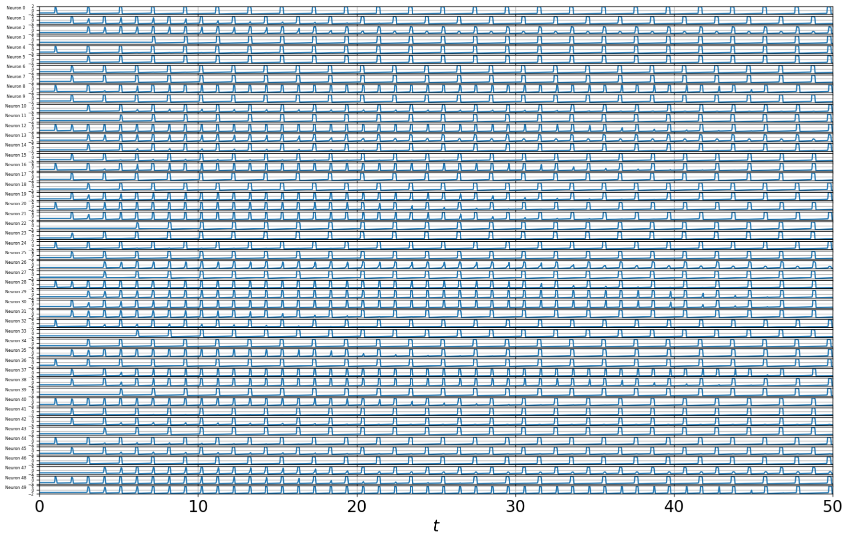

Figure 2.7.6: Simulation on the Barabasi-Albert network, after applying the Bipartition Algorithm: the system evolves towards bipartite synchrony.

2.8 Robustness of the main random network architectures

Having looked at the effects on the dynamics of our bipartition procedure, it is useful and interesting to look at how the various network architectures are modified in themselves by algorithm. This consideration can be highly informative since it tells us, indirectly, which features of a network are more sensitive to bipartition using this algorithm. In other words, it tells us whether the algorithm reasonably preserves the initial structure, or if it destroys it. To test how much the effects on the observables are related to the aimed nature of the removals operated by the algorithm, we compare their variation in the bipartition process with the one observed for a randomised removal of the same amount of edges. The randomised removal is performed taking care not to disconnect the network in the process. A relevant observable in the following analysis will be the Average Shortest Path Length (ASPL), that is the average of the lengths of all the shortest paths connecting distinct pairs of nodes on the network.

Erdős-Renyi network We first test the algorithm on the simplest random network architecture. We consider an instance of Erdős-Renyi network with 100 nodes, with attachment probability $p = 0.25$. We apply the algorithm, which removes 458 links, out of the initial 1210. To assess the measure to which the network structure has been affected, we look at the degree distribution before and after applying the algorithm, Figure 2.8.1. The distribution appears peaked around a lower value, as is attested by the mean degree, that decreases from 24.2 to 15.04[3]. The sample standard deviation decreases as well from 3.7 to 3.2, showing a closer spread of the degrees in the bipartite network. The average shortest path length (ASPL) increases slightly, from 1.76 to 2.21, as one would expect, since before bipartition, the network contains intra-partition links that act as shortcuts, with respect to the bipartite configuration, and that are removed, by definition, by the bipartitition algorithm. To test how much the aimed removal of the bipartition process affects these observables, we evaluate them for a randomised removal of the same number of edges as those removed by the algorithm. We obtain a mean degree of 15.0 ± 3.4[3], and an ASPL of 1.94. The evidence considered points to the fact that the Erdős-Renyi network structure is modified but not entirely destroyed by the bipartition process.

[3] Both in the algorithmic and in the randomised removal, the decrease of the average degree amounts to ≈ 38%, a value close to the percentage of removed links.

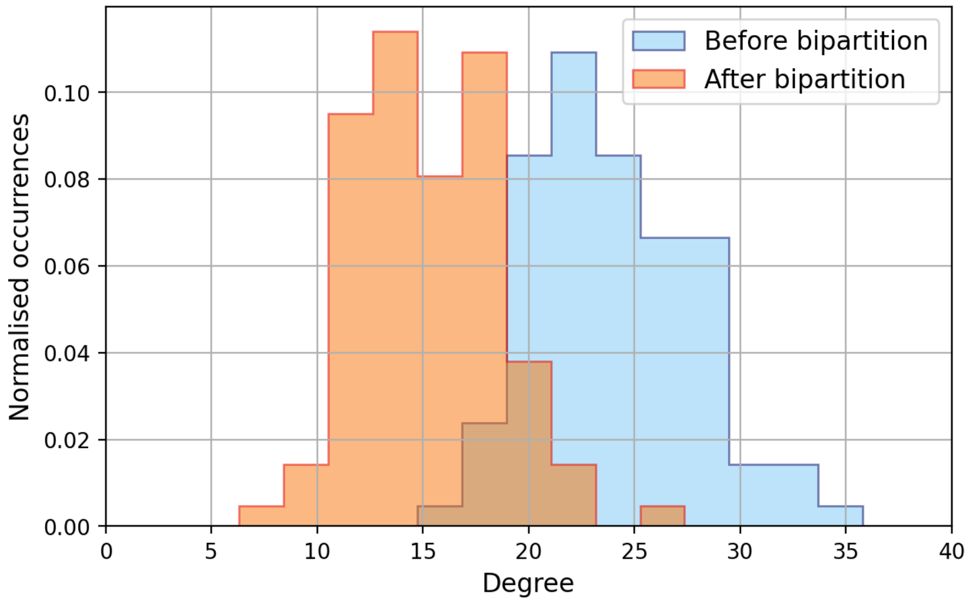

Figure 2.8.1: Degree distribution before and after application of the bipartition algorithm. The initial network is an Erdős-Renyi graph with 100 nodes and 1210 edges, the final network having 752 remaining edges. The histogram contains 20 bins from 0 to 40.

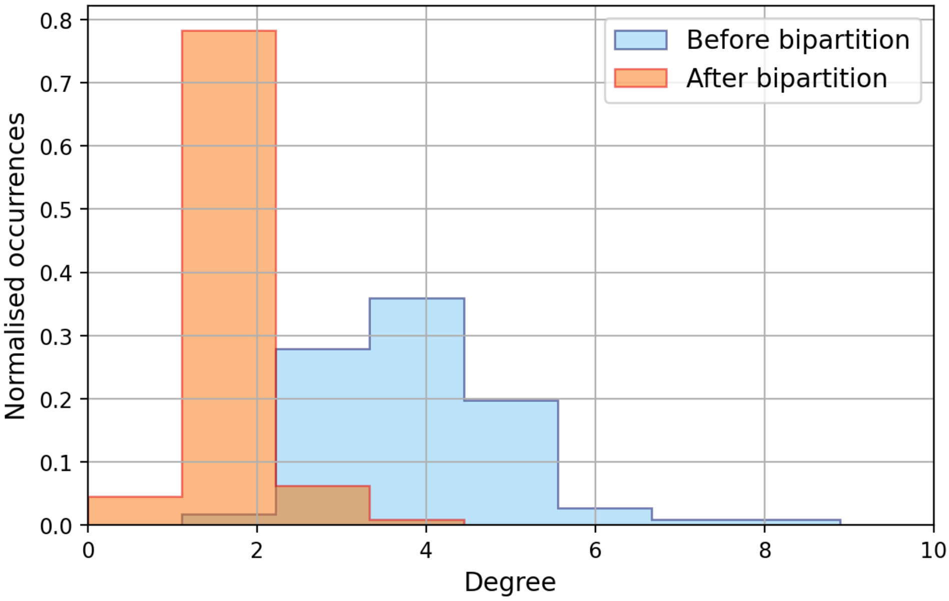

Figure 2.8.2: Degree distribution before and after application of the bipartition algorithm. The initial network is a Watts-Strogatz graph with 100 nodes and 200 edges, the final network having 102 remaining edges. The histogram contains 10 bins, from 0 to 10.

Watts-Strogatz network We test the algorithm on a Watts-Strogatz network as well, to test the robustness of small-world property against the bipartition algorithm. The sample network we consider has 100 nodes and 200 edges, built with an initial nearest-neighbour coordination number of 4, and with a rewiring probability of $p = 0.25$. The bipartition algorithm removes 98 edges. By looking at the degree distributions before and after bipartition, Figure **2.8.2**, we notice here as well a diminished average degree connectivity, from 4 to 2.04, and a sharper nature of the distribution peak, as noticeable from the sample standard deviation, which changes from 0.99 to 0.40. The Average Shortest Path length is a very relevant observable for this class of random graphs, since a small average shortest path length is part of the definition of the small-world property. In the numerical experiment,we observe a dramatic increase in the average shortest path length, which amounts to 3.98 before bipartition, and to 32.35 after applying the algorithm. If we compare again these values with those obtained by removing an equal number of edges at random, we observe that the average degree 2.040 ± 0.916 is analogous to that obtained via bipartition, albeit with a larger spread of values, while the ASPL, evaluated to be 9.51, is larger

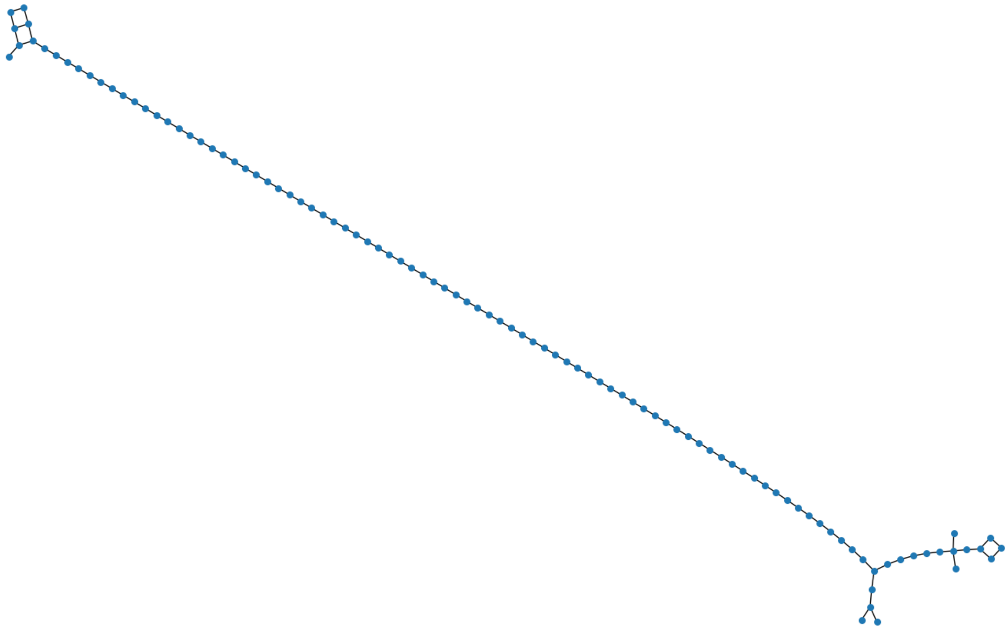

Figure 2.8.3: Visualisation of the bipartite graph obtained from the Watts-Strogatz network. Notice the chain-like structure, with occasional branching points, and small cycles of four nodes at the extremities.

than its initial value, but has not increased as dramatically. These numbers hint to a destruction of the small world property as a result of the application of the algorithm, and indeed, if we visualise the final graph by means of a force-directed network visualisation algorithm [KK89], we obtain Figure 2.8.3.

The network structure appears to become chain-like, with a small number of branching points. At some of the extremities of the structure, such as at the inferior right and superior left corner of Figure 2.8.3, we observe the presence of closed cycles of length 4. This has been a common feature in all the bipartition runs we have performed starting from Watts-Strogatz graphs. A dynamical interpretation can be attempted for their appearance. Indeed these structures are able to produce a self sustained bipartite oscillation pattern. A single cycle of length four is in fact a bipartite structure: if we arrange the cycle as a square we can consider as partitions the two pairs of opposing corners, the self-sustained bipartite oscillation consists of an alternate firing of these two pairs. In the context of a chain, these structure play the role of pacemakers and of self-sustained spike producers, that are able to send action potentials indefinitely and continuously along the chain.

Barabási-Albert network Finally, we test the algorithm on a Barabási-Albert network. This network model has been designed to display the scale free property, that corresponds to a power-law degree distribution. The Barabási-Albert model attains this degree distribution as the result of a preferential attachment growing process. The process begins from an undirected network of m_0 nodes. For each step, a node is added to the network, and connected to $m \leq m_0$ pre-existing nodes. Node i is selected to be one of the m nodes with a probability

$$\mathbb{P}(i) = \frac{d_i}{\sum_j d_j}$$

so that nodes with high degree are more likely to increase their connectivity even more, while poorly connected nodes are unlikely to gain a significant amount of new connections. This is the preferential attachment building rule which generates the scale-free behaviour. In particular for the Barabási-Albert model, the degree distribution follows a power law with exponent -3.

$$\mathbb{P}(d) \sim d^{-3}$$

For our numerical experiment we consider a Barabási-Albert network of 100 nodes, built with a value of $m = 4$. The network initially contains 384 edges, containing only 105 after bipartition. The degree distribution before and after the algorithm application is presented in Figure 2.8.4. By looking at the initial distribution, the blue histogram in Figure 2.8.4, we observe the typical shape of a power-law, with a right tail that becomes thin but non-vanishing for high degrees. Conversely, we observe that the degree distribution for the resulting bipartite graph is sharply peaked on $d = 2$. Similarly to the previous Watts-Strogatz case, the Average Shortest Path Length increases from 2.35 to 32.4 in the bipartition process, while, under random removal of edges, it only grows as large as 6.87. These observations are sufficient to suppose that the scale-free nature of the network is destroyed in the bipartition process. In fact, if we visualise the bipartite network obtained from the Barabási-Albert model, we obtain Figure 2.8.5 analogously to the Watts-Strogatz case, the original scale-free structure has been completely destroyed. The graph looks like a chain with some branching points. At the upper right corner of the image, we can observe a cluster, of cycles of length 4. The cycles are intertwined so that no cycle of odd length is created. This observation is in accord with previous observations, such as in [Kan+11], that the only multipartite synchronies that can arise on a network, are those which are made up of a number of clusters equal to the Greatest Common Divisor of the length of the network loops.

We perform the mentioned tests on ten network instances for each of the three models, so to reduce the impact of fluctuations in the previously discussed

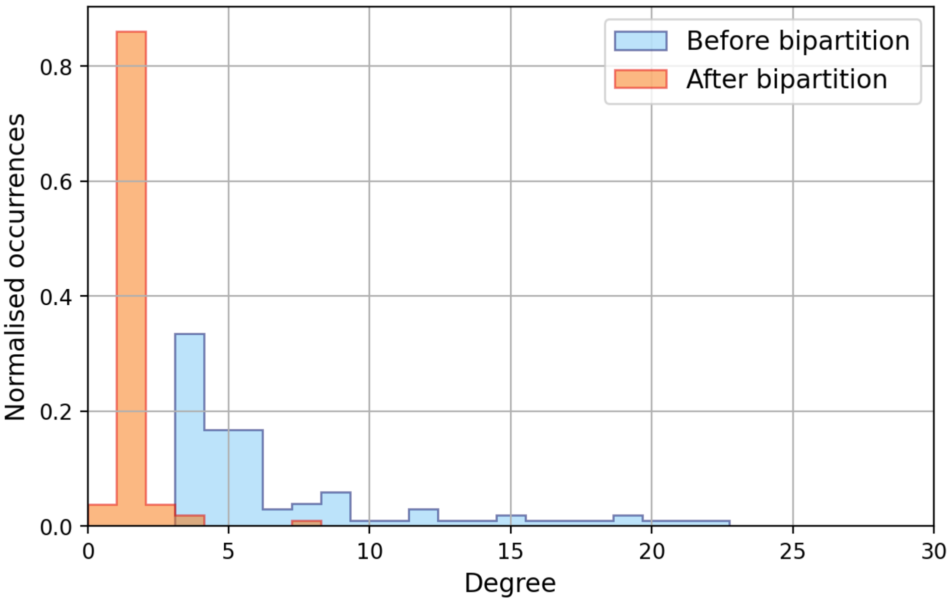

Figure 2.8.4: Degree distribution before and after application of the bipartition algorithm. The initial network is a Barabási-Albert graph with 100 nodes and 384 edges, the final bipartite network has 105 edges. The histogram contains 30 bins from 0 to 30.

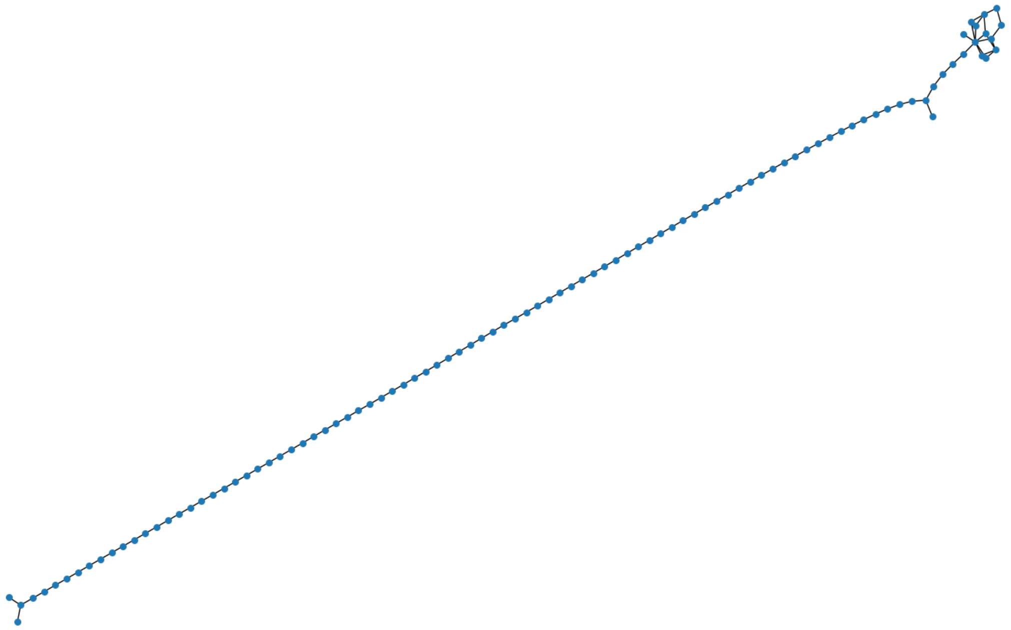

Figure 2.8.5: Force-directed visualisation of the bipartite graph obtained from a Barabási-Albert graph. Similarly to the Watts-Strogatz case, the original structure is largely destroyed, and the network structure can be described in terms of a chain with branching points.

Network type	$n_{initial}$	$n_{bipartite}$	$ASPL_{initial}$	$ASPL_{bipartite}$	$ASPL_{random}$
Erdős-Renyi	1232 ± 27	474 ± 24	1.752 ± 0.006	2.200 ± 0.013	1.932 ± 0.25
Watts-Strogatz	200	97.6 ± 1.2	3.929 ± 0.075	29.824 ± 2.255	10.472 ± 1.209
Barabási-Albert	384	280 ± 3.7	2.374 ± 0.017	32.431 ± 1.193	7.114 ± 1.578

Table 2.1: Statistics for the mentioned observables, over a sample of ten networks for each network type. Values are expressed as mean ± sample standard deviation. $ASPL_{initial}$, $ASPL_{bipartite}$ and $ASPL_{random}$ indicate respectively the initial Average Shortest Path Length, its value after bipartition and its value after randomised removals. $n_{initial}$ and $n_{bipartite}$ represent the number of edges before and after bipartition. Notice that $n_{initial}$ has no uncertainty for the W-S and B-A models, since the number of links is fully determined by the constructive properties.

observations. The results are presented in Table 2.1, and are in accord with the evidence just presented for single network instances. The bipartition process causes an increase of the Average Shortest Path Length, which for the Erdős-Renyi network is comparable to the one caused by randomised removals, while for the Watts-Strogatz and Barabási-Albert models it is considerably larger.

2.9 Chapter summary and concluding remarks

From the analysis presented in this chapter, we are able to conclude the following:

- In the in-degree-normalised scheme, the completely synchronous dynamics is always stable, unless the underlying network is bipartite.
- In such a case, the stable state for the collective dynamics is a bipartite one, where two clusters fire in an alternating manner.
- If the network is bipartite, the partitions, i.e. the clusters of simultaneously firing neurons, are those identified by the vectors

$$\mathbf{1}_P = \frac{1}{2}(\mathbf{1}_N + \mathbf{a}_1) \quad \mathbf{1}_Q = \frac{1}{2}(\mathbf{1}_N - \mathbf{a}_1)$$

  in the sense that node i belongs to partition P if the i-th entry of $\mathbf{1}_P$ is non zero, to Q if the i-th entry of $\mathbf{1}_Q$ is non zero.
- Given an undirected network, we can make it bipartite by applying the algorithm described in Section 2.6 and Appendix D, which acts by selecting edges via perturbative calculus and removing them until the network becomes bipartite.

- The algorithm tends to destroy the small-world and scale-free properties, on the tested random models, while it does not affect much the structure of Erdős-Renyi networks.

- The possibility of introducing multipartite states with more than two components, e.g. by moving the second-largest Laplacian eigenvalue towards 2 as well, while keeping the network connected is ruled out by Theorem 1.2.3. Indeed, once the network is bipartite the Laplacian spectrum is symmetrical w.r.t. 1, so any structural change that drives the second-largest eigenvalue to 2, simultaneously has the Fiedler value collapse to 0, thus disconnecting the graph.

- Since the transition from a completely synchronous to a bipartite synchronous state can be essentially regarded as a perioud-doubling dynamical transition, the previous conclusion also rules out the possibility of a period-doubling route to chaos being attained via network modification.

A number of questions arise from what we have just presented. A natural one would be, for example, to consider the *optimality* of the Bipartition algorithm, investigating how the number of removed links scales with the size of the network, and how it depends on the initial network structure. Does the algorithm cut the minimal set of links that have to be removed in order to make the network bipartite, or considering other strategies (e.g. considering the cutting of several edges at a time) could it do better? As we have seen, moreover, the algorithm tends to destroy the scale-free or small-world properties of a network, another interesting continuation would be to look at specific strategies that could preserve these network features. Finally, the application to directed networks, with the necessary modifications to the algorithm is another topic worthy of exploration.

Chapter 3

Synchronisation phenomena in the out-degree normalised scheme

The picture is quite different when one considers the out-degree normalisation scheme. Indeed in this case $\mathcal{A}$ enjoys a unit-column-sum condition, and therefore, does not admit $\mathbf{1}_N$ as a right eigenvector any longer. Nonetheless, we still work self-consistently, even if it will become immediately evident that this approach can only be an approximation.

3.1 Self-consistent approach

Let us recall the dynamical equations for the system.

$$\begin{aligned} \varepsilon \dot{u}_i(t) &= u_i(t) - \frac{u_i^3(t)}{3} - v_i(t) + C \sum_{j=1}^{N} \mathcal{A}_{ij} \left[u_j\left(t-\tau\right) - u_i\left(t\right) \right] \\ \dot{v}_i(t) &= u_i(t) + a \end{aligned} \tag{3.1}$$

where now $\mathcal{A} = AD^{-1}$ is normalised on the right, i.e. by columns. We proceed to insert the usual self-consistent hypothesis, that at synchrony all neurons evolve along the same trajectory over time

$$u_1(t) = u_2(t) = \ldots = u_N(t) \equiv u_s(t), \quad v_1(t) = v_2(t) = \ldots = v_N(t) \equiv v_s(t) \tag{3.2}$$

into the equation, thus obtaining, for each node

$$\begin{aligned} \varepsilon \dot{u}_s(t) &= u_s(t) - \frac{u_s^3(t)}{3} - v_s(t) + C F_i \left[u_s\left(t-\tau\right) - u_s\left(t\right) \right] \\ \dot{v}_s(t) &= u_s(t) + a \end{aligned} \tag{3.3}$$

The approximate nature of this approach is immediately highlighted by an absurd: the self-consistent hypothesis implies that the self-consistent solution will be *different for each node*. This is dictated by the presence of the F_i, which we name the *self-consistent factor*, that taking into account the normalisation scheme amounts to

$$F_i = \sum_j \mathcal{A}_{ij} = \sum_{j \sim i} \frac{1}{d_j} \tag{3.4}$$

where by $j \sim i$ we mean that j is running over the set of nodes which are adjacent to node i. It is the sum of the weights of all the incoming links. From an effective point of view, it determines a scaling of the coupling intensity felt by node i. It is on this consideration that we base our approach.

The idea that justifies this approach, is that considering *neuronal* synchrony, we are interested only in the simultaneous *firing* of the units, i.e. simultaneous above-threshold activity, and not about the magnitude of such activity, as long as it is above-threshold. Within this approximation, therefore, we expect that the onset of the fully synchronised phase will be related to the number of nodes for which $F_i C > C^*(\tau)$. We will be able to consider this result as an upper bound on the minimal coupling strength for synchronisation.

3.2 Example on a small graph

Let us consider, as an example, the small undirected network in Figure 3.2.1, where the edge weights are understood to be those established by the out-degree-normalisation scheme. Nodes 0 through 5 have degree 5, and nodes 6 through 11 have degree 3. This yields a self-consistent factor of

$$F_i = \begin{cases} \dfrac{17}{15} & \text{if} \quad 0 \leq i \leq 5 \\ \\ \dfrac{13}{15} & \text{if} \quad 6 \leq i \leq 11 \end{cases} \tag{3.5}$$

Therefore the most connected nodes are favoured to sustain a synchronous solution, as the effective value of the coupling constant they perceive is amplified by a $\frac{17}{15}$ factor. In this sense, we expect the center ring to be able to support self-sustained firing around the value of $\frac{15}{17}C^*(\tau)$.

We simulate the system, with parameters $\tau = 1$, $\varepsilon = 0.01$, $a = 1.3$, varying C. To center our analysis around the value $C^*(\tau)$ we introduce the value

$$r = \frac{C}{C^*(\tau)}$$

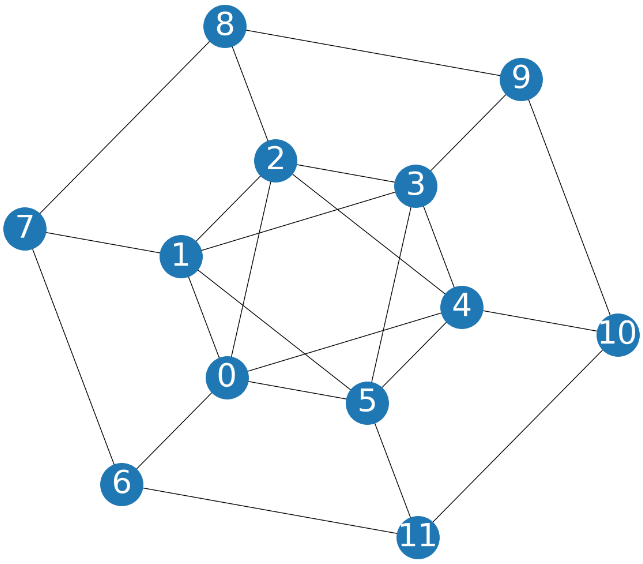

Figure 3.2.1: Small graph example.

so that the threshold value be fixed at $r = 1$. For $r \geq \frac{15}{17}$ we observe full synchronisation throughout the network. As r is lowered, the completely synchronous activity persists, down until $r \approx 0.635$, below which self-sustained activity ceases. These findings allow us to highlight another feature of the approach we have undertaken so far. Within this normalisation scheme the effects of redundancy provided by the graph structure, are more relevant than in the previous case. Indeed, a possible explanation for this further lowering of the full synchronisation threshold, could be that as synchronous, largely self-sustained, clusters emerge within the network, they start exerting on more peripheric nodes an effect more akin to an external forcing, rather than interacting with them in a mutual manner. This hypothesis can be supported by simulating a driver-peripheral configuration, studying the peripheral activity versus the coupling strength. The results in Figure 3.2.2. show that the one-way coupling threshold for triggering activity in a peripheral neuron is much lower than the self-coupling value needed for self-sustained activity in the feedback scheme of Figure 2.1.1. To illustrate the regimes observed on this small network, we present the network time series for several values of r in Figures 3.2.3, 3.2.4 and 3.2.5. The initial pulse has been administered on the nodes with higher-than-average degree, which are the most favoured ones, as we established with the previous analysis.

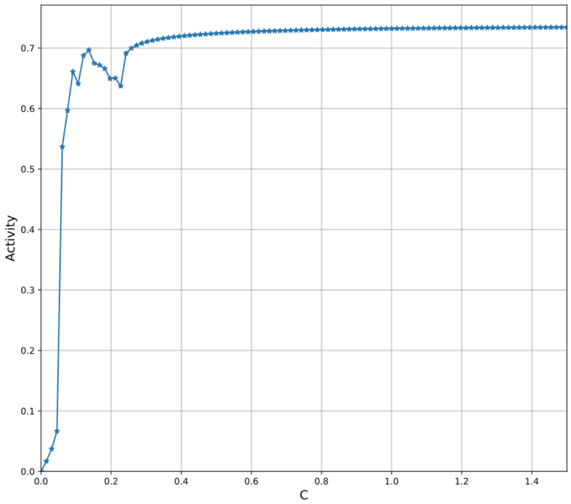

Figure 3.2.2: Plot of peripheral neuron activity versus the driver-peripheral coupling strength, activity is evaluated as in the case of Figure 2.1.1. In this simulation, the driver is a FHN neuron with feedback, with parameters $\tau = 1$, $\varepsilon = 0.01$, $a = 1.3$, $C_{feedback} = 1$.

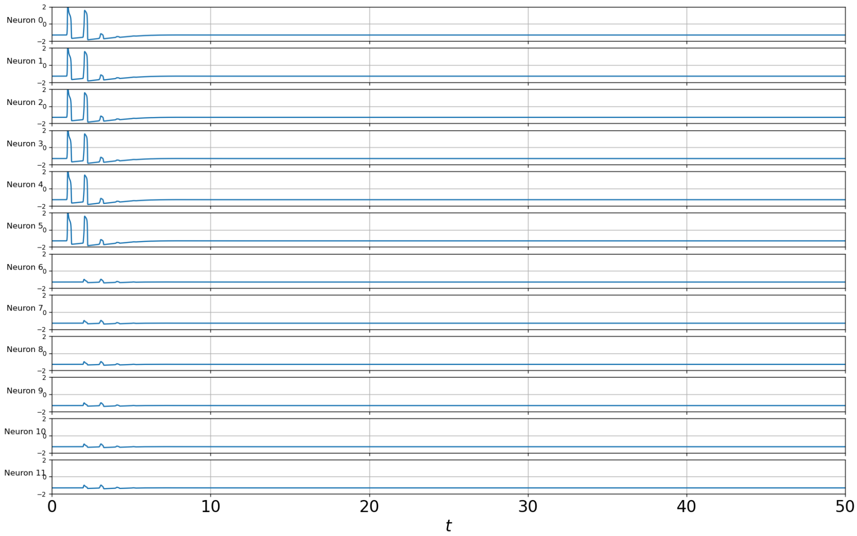

Figure 3.2.3: Simulation with $r = 0.5$. After a very quick transient, all activity vanishes from the network.

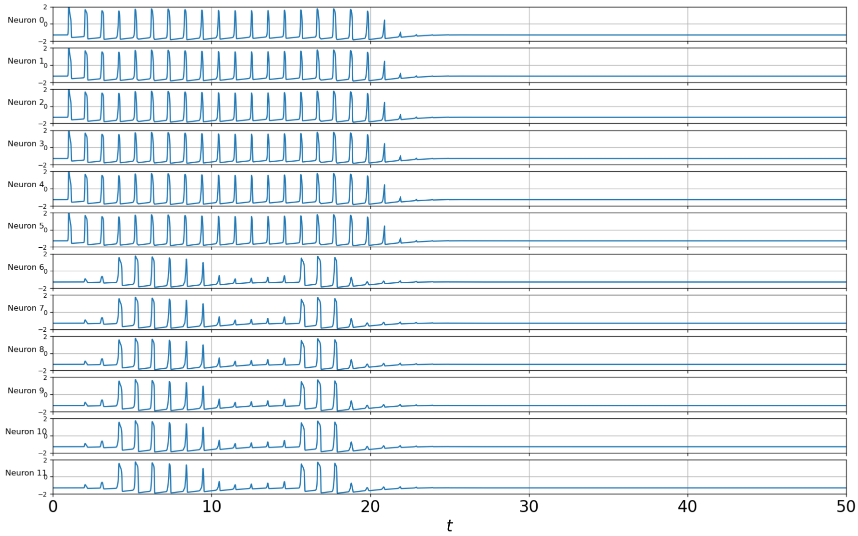

Figure 3.2.4: Simulation with $r = 0.6325$. The transient with network activity becomes longer, lesser connected neurons show bursting effects.

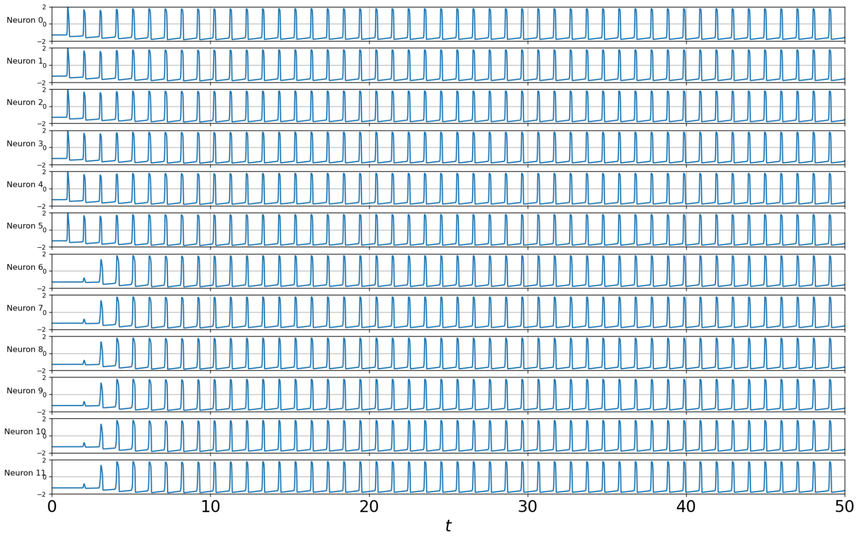

Figure 3.2.5: Simulation with $r = \frac{15}{17}$. The network synchronises fully.

3.3 Mean Field Approximation

In random models F_i becomes a random variable, in principle with a complicated distribution. We simplify the treatment of the problem by introducing a Mean Field Approximation. This implies assuming that each node is connected to a number of other nodes equal to the average degree $\langle d \rangle$, neglecting the connectivity fluctuations that may be present. Due to this approximation, each term of the sum in the definition (3.4), contributes with a fixed amount $\frac{1}{\langle d \rangle}$, and we must consider on average $\langle d \rangle$ contributions, thus obtaining a Mean Field Value for the self-consistent factor $\bar{F}$ equal to one.

$$\bar{F} = 1$$

Notice that this approach is exact on regular graphs, and we expect it to work best on models with small connectivity fluctuations, the result progressively worsening as the fluctuations grow more relevant. We proceed to compare the approach with simulations. The dynamical parameters of choice for simulations will be the usual $a = 1.3$, $\varepsilon = 0.01$, $\tau = 1$. The initial perturbation provided to initiate activity will be given in the form of Gaussian pulses, delivered on the nodes with higher than average connectivity.

Erdős-Renyi network We first reconstruct the probability distribution of the F_i by a Monte Carlo approach, calculating it for an E-R network with $p = 0.25$ and 15000 nodes. We obtain Figure 3.3.1, which shows a peaked distribution on the mean field value $\bar{F} = 1$, which, as it is shown in the picture, fits well with a Gaussian distribution, providing evidence for a central limit theorem behaviour, reasonably due to the great average regularity of the Erdős-Renyi model. We proceed to trace back the Central Limit Theorem behaviour to the independence of degrees for different nodes.

The degree of each node in an Erdős-Renyi network is independent from that of other nodes, and by construction it is the result of a success-fail process, where $N - 1$ attachment attempts are performed, one for each of the other nodes. Each attempt succeeds with probability p, or fails with probability $q \equiv 1-p$. As a result, the degree of each node is described by a binomial distribution with average value $(N-1)p$. When treating networks with a large number of nodes, if the average degree $(N-1)p \approx Np \equiv \mu$ is kept fixed, the degree distribution can be approximated in a Poisson form (3.6).

$$\mathbb{P}(d) = \frac{\mu^d}{d!} e^{-\mu}, \quad d = 0, 1, 2, \ldots \tag{3.6}$$

With these definitions in place, the definition for the self-consistent factor (3.4), contains, on average, a sum of μ independent, identically distributed random variables. By the central limit theorem we know that if each of these variables has finite variance, for a large enough number of them the distribution of their sum is well approximated by a Gaussian. We reasonably expect that the order of magnitude of the number of terms in the sum be proportional to the average degree μ of the network, and so we expect this approximation to be better for large μ. We set out to prove that $\frac{1}{d}$ has a finite mean and variance, when one considers a large Erdős-Renyi network. In order to avoid divergences, we restrict the domain of the probability mass function to positive d. Notice that this restriction is implied by the definition (3.4), since no node can have a neighbour with degree zero, i.e. which has no neighbours in turn.[1] This is equivalent to state that the distribution of $\frac{1}{d}$ is conditioned by the event $d \geq 1$, so that we must normalise the previous expression with $\mathbb{P}(d \geq 1)$, obtaining expression (3.7)

$$\mathbb{P}_c(d) \equiv \mathbb{P}(d|d \geq 1) = \frac{\mathbb{P}(d)}{\mathbb{P}(d \geq 1)} = \frac{1}{1-e^{-\mu}}\frac{\mu^d}{d!}e^{-\mu}, \quad d = 1, 2, 3, \ldots \tag{3.7}$$

where the c subscript stands for "connected", meaning that $\mathbb{P}_c$ is the probability mass function for the node degree, provided that the node is connected, we shall denote expectations taken using this probability mass function as $\langle \cdot \rangle_c$.

The expression for the mean value reads

$$\left\langle \frac{1}{d} \right\rangle_c = \frac{1}{1-e^{-\mu}} \sum_{d=1}^{\infty} \underbrace{\frac{\mu^d}{d!}\frac{e^{-\mu}}{d}}_{T_{1,d}} \tag{3.8}$$

by introducing the ratios $R_{1,d}$ and taking the limit for $d \to \infty$

$$R_{1,d} = \left| \frac{T_{1,d+1}}{T_{1,d}} \right| = \frac{\mu}{d}\left(1 + \frac{1}{d}\right)^{-2} \xrightarrow[d\to\infty]{} 0 \tag{3.9}$$

we are able to prove the absolute convergence of the series by the D'Alembert ratio criterion, since the ratio tends to a value lower than 1. We have thus proved that the average value $\left\langle \frac{1}{d} \right\rangle_c$ is finite. We move on to the variance, and observe that the usual relation holds also for the variance of the inverse, (3.10).

$$\mathrm{Var}\left[\frac{1}{d}\right] = \left\langle \left(\frac{1}{d} - \left\langle \frac{1}{d} \right\rangle_c\right)^2 \right\rangle_c = \left\langle \frac{1}{d^2} \right\rangle_c - \left\langle \frac{1}{d} \right\rangle_c^2 \tag{3.10}$$

[1] Erdős-Renyi networks are connected almost surely (with probability 1) if $p > \frac{(1+\varepsilon)\log(N)}{N}$. For fixed large N, this corresponds to fixing $\mu > (1+\varepsilon)\log(N)$. For example, the condition is verified for the graph used to build Figure 3.3.1, since we have $0.25 > \log\big(1.5 \times 10^4\big)/(1.5 \times 10^4) \approx 6.4 \times 10^{-4}$.

Since we have just ascertained the finiteness of the mean value of the inverse, to prove that the variance is finite we only need to consider the second negative absolute moment. Its expression reads

$$\left\langle \frac{1}{d^2} \right\rangle_c = \frac{1}{1-e^{-\mu}} \sum_{d=1}^{\infty} \underbrace{\frac{\mu^d}{d!} \frac{e^{-\mu}}{d^2}}_{T_{2,d}} \tag{3.11}$$

we analogously build the ratio, which in the limit yields

$$R_{2,d} = \left| \frac{T_{2,d+1}}{T_{2,d}} \right| = \frac{\mu}{d} \left(1 + \frac{1}{d}\right)^{-3} \xrightarrow[d\to\infty]{} 0 \tag{3.12}$$

so that the variance converges as well, by the D'Alembert ratio criterion. Therefore, the Central Limit Theorem applies, and for large networks F_i is distributed around the Mean Field Value $\bar{F} = 1$ according to a Gaussian distribution. In Appendix E, we are able to obtain a leading order estimate in the case of large N and large μ as

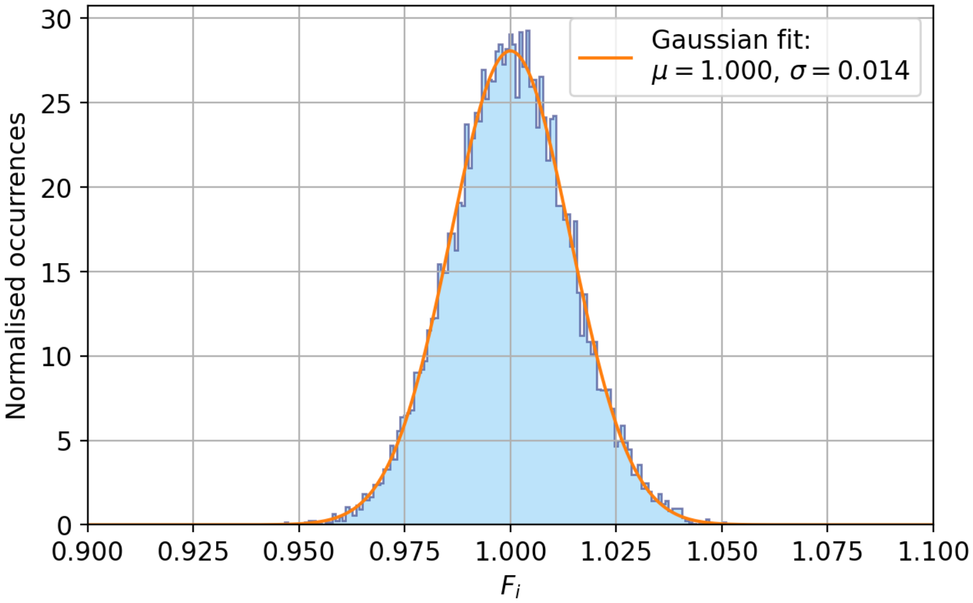

Figure 3.3.1: Self-consistent factor distribution for an E-R network with 15000 nodes, $p = 0.25$. The histogram contains 250 bins from 0.9 to 1.1. The orange line shows a Gaussian fit.

$$F_i \sim \bar{F} + \frac{\mathrm{G}(0,1)}{\sqrt{Np}} = 1 + \frac{\mathrm{G}(0,1)}{\sqrt{Np}} \tag{3.13}$$

where G$(0,1)$ is the standard Gaussian random variable of zero mean and unit variance. This approximation gives an explanation for the very small standard deviation observed in the fit [2] in Figure 3.3.1. Therefore, for a large network, the mean value dominates over the fluctuations, and the smaller the resulting variance, the better the Mean Field Approximation will describe the network synchronisation.

To test our findings we simulate the dynamics on an Erdős-Renyi random network with $N = 50$ nodes, and attachment probability $p = 0.25$. The small size of the simulated network, in comparison to the one considered in the Monte Carlo experiment is due to computing power limitations. Nonetheless, we expect the simulation results to reproduce our findings to some degree, even if at this scale the connectivity fluctuations are still relevant and will alter the results, with respect to the predictions for large N. What we observe from the simulation, is that at low r values, Figure 3.3.2, the network does not support self-sustained activity. As r is increased, a completely synchronised fraction of nodes emerges, Figure 3.3.3, and by $r = 1$, the synchronisation is full, all over the network, Figure 3.3.4. Therefore, for the E-R network, this Mean Field approach appears to provide a reasonable upper bound for the minimal coupling strength needed to observe synchronisation. Nonetheless, the network is actually largely synchronised for values as low as $r = 0.7$, a fact that can be interpreted as an effect of the fluctuations, that on a small network are still relevant, but also with the observations made in Section 3.2, concerning the transition from an interaction dynamics, to a driver-peripheral one, once large enough clusters of the network have synchronised.

Watts-Strogatz network Next, we study the case of a Watts-Strogatz network. As for the E-R case, we build the distribution of F_i numerically, from a Watts-Strogatz network with 15000 nodes, with constructive parameters of $k = 4$ initial nearest neighbours per node and $p = 0.25$ rewiring probability. The distribution, Figure 3.3.5, is still peaked around 1, but has two larger tails, compared to the previous case. These facts hint at a reasonably sound prediction of the transition in terms of the Mean Field Approximation, albeit probably worse than in the Erdős-Renyi case due to the larger spread of the F_i distribution. As we did previously we simulate a network with the same constructive parameters $k = 4$ and $p = 0.25$, but with a reduced number of nodes $N = 50$ due to computing limitations. This time as well, we expect the simulation to qualitatively reproduce the behaviour hinted at by the numerically reconstructed distribution, albeit suffering from the drawbacks of a very

[2] For comparison $\frac{1}{\sqrt{15000 \cdot 0.25}} \approx 0.016$, which is reasonably close to the observed 0.014.

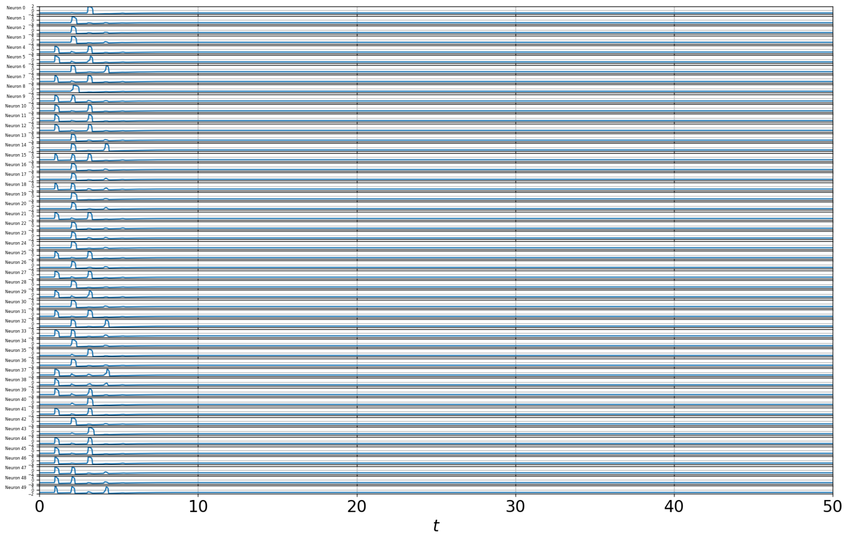

Figure 3.3.2: Activity plot for the E-R network at $r = 0.5$.

small system size. The simulations show a picture similar to the E-R one, but with slight differences. For r as low as 0.5, no activity can sustain itself over the network, Figure 3.3.6. As r grows, some portions of the network start to synchronise, Figure 3.3.7, although at a smaller rate than in the Erdős-Renyi case. As we reach $r = 1$, the synchrony is full, Figure 3.3.8.

Barabási-Albert network Finally, we consider a Barabási-Albert network. As for the previous cases, we build the distribution of F_i numerically, from a Barabási-Albert network with 15000 nodes, with attachment parameter $m = 4$. This time, as we expect, the distribution, in Figure 3.3.9 is strikingly different from both of the previous cases. The histogram shows a peak below the mean field value $\bar{F} = 1$, and a power-law tail, degrading above 1. This is not unexpected, as a favourable self-consistent factor ($F_i < 1$) arises for a high-degree node, when it is connected to many lower-degree nodes, a condition that is realised for only a few of the nodes in a B-A model, most of them being connected to nodes with degrees higher than their own, due to the preferential attachment dynamics present in the model growth process. Moreover, we cannot expect a Central Limit Theorem behaviour to set in, in this case, as the preferential attachment construction introduces correlation in the node degrees. In addition to this, even if we were able to consider the degrees of neighbouring nodes as independent, the power-law behaviour of the degree distribution pre-

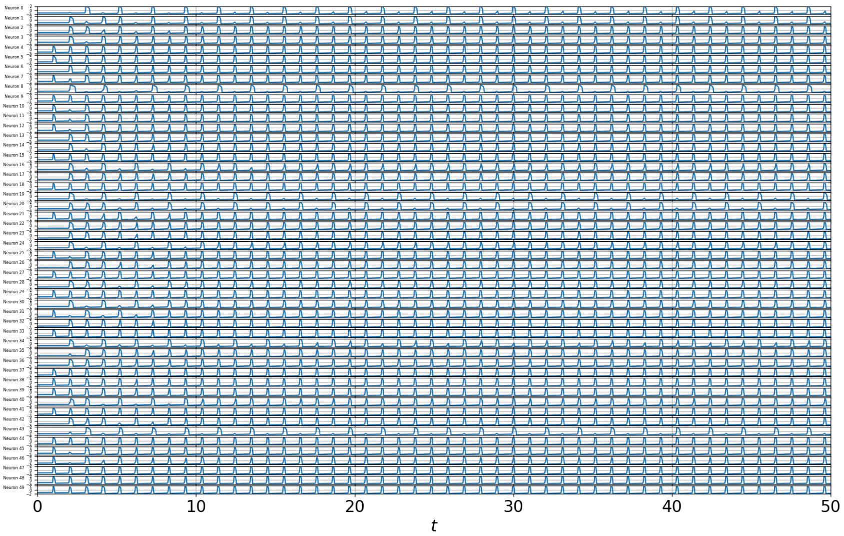

Figure 3.3.3: Activity plot for the E-R network at $r = 0.7$. Notice e.g. neurons 0 and 1, which are following a bipartite-like trajectory, even though most of the network is synchronised.

vents the variance from being finite, thus hindering the possibility of a Central Limit Theorem approach. All these facts suggest a poor description of the transition, if it will be present, in terms of the Mean Field Approximation, which we investigate with simulation on a smaller network. As for the previous cases we simulate the neuronal dynamics on a network with $N = 50$ nodes and attachment parameter $m = 4$. In analogy to the previously tested models, we expect the smaller network simulation to bear at least some resemblance to the behaviour we would observe on a larger one, of a size comparable to the one used in the Monte Carlo reconstruction. The simulations, indeed, show a very different scenario with respect to the other cases. For $r = 0.5$ we have no self-sustained activity, as in the previous cases, Figure 3.3.10. This time, though, even if we raise r to the mean field value $r = \bar{F} = 1$ we cannot bring the network to synchronise, Figure 3.3.11, but only obtain some scattered, non-synchronised activity. If we try to increase r above the $\bar{F}$ value, e.g. to $r = 1.5$, activity ceases again, Figure 3.3.12. This could be understood in terms of the phenomenon we observed also in the single neuron with feedback case, in Figure 2.1.1, where a too intense coupling makes the system almost insensitive to external stimuli.

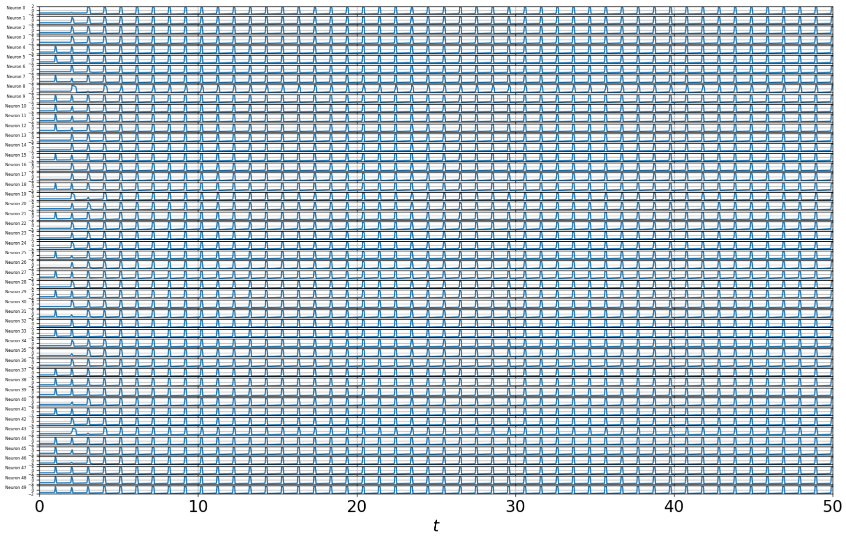

Figure 3.3.4: Activity plot for the E-R network at $r = 1.0$. Synchronisation is full, all over the network.

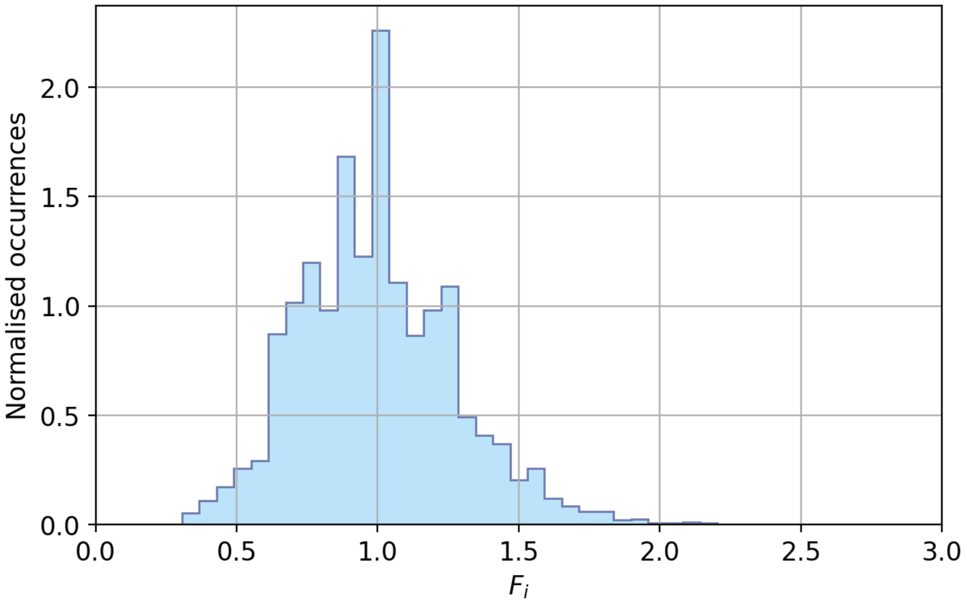

Figure 3.3.5: Self-consistent factor distribution for a Watts-Strogatz network with 15000 nodes, $k = 4$, $p = 0.25$. The histogram contains 50 bins from 0 to 3.

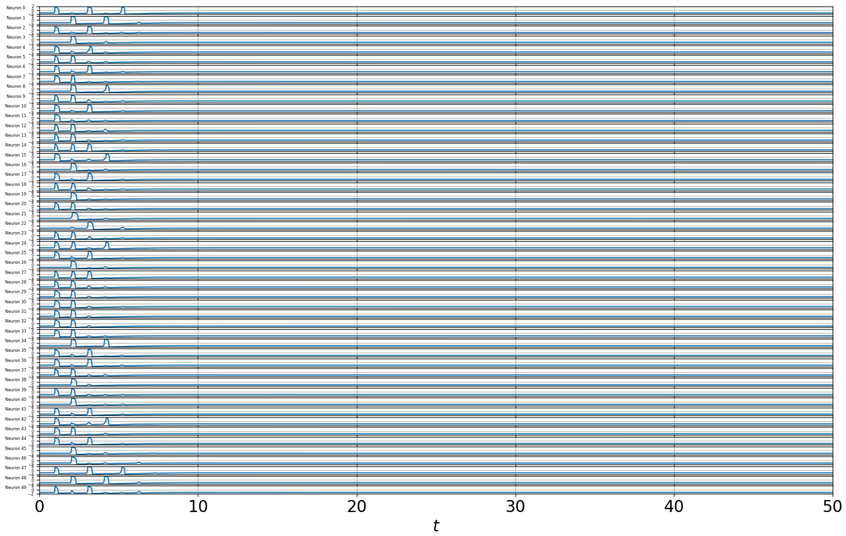

Figure 3.3.6: Activity plot for the Watts-Strogatz network at $r = 0.5$.

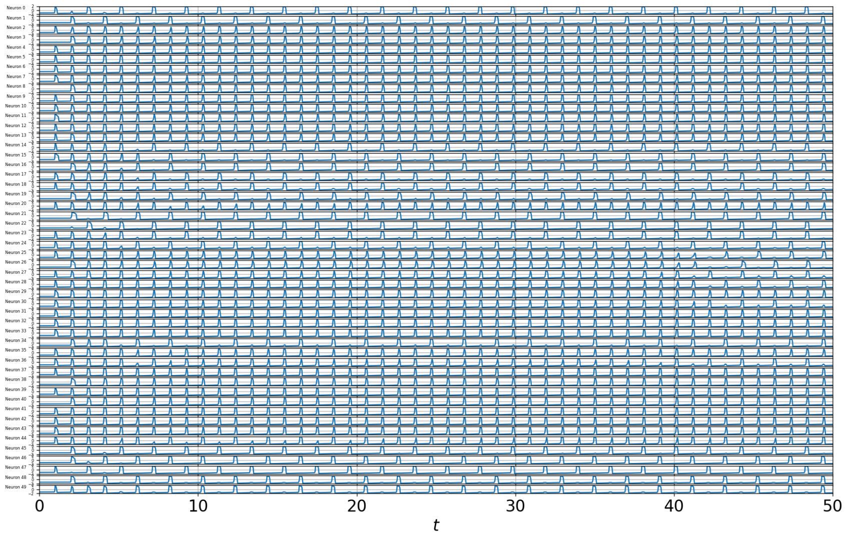

Figure 3.3.7: Activity plot for the Watts-Strogatz network at $r = 0.75$. Compared to the E-R case of similar value (Figure 3.3.3), we see that smaller portions of the network have synchronised.

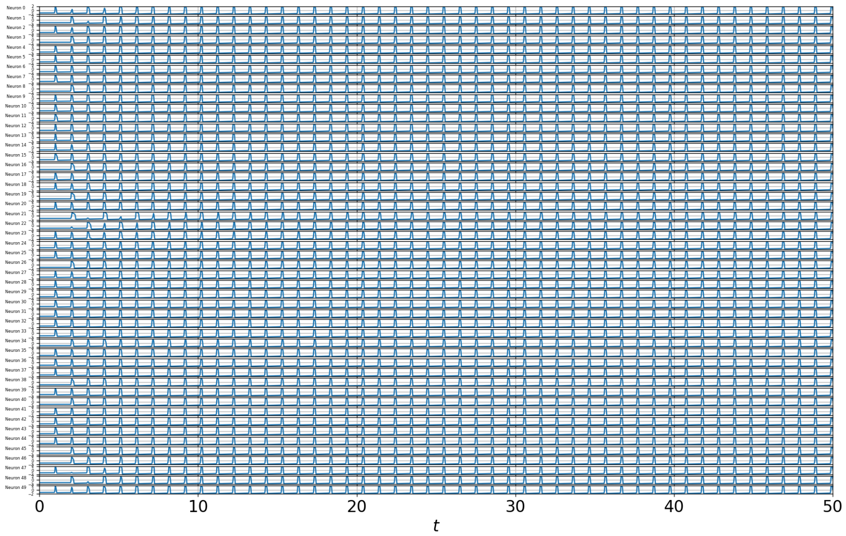

Figure 3.3.8: Activity plot for the Watts-Strogatz network at $r = 1.0$. The network is fully synchronised.

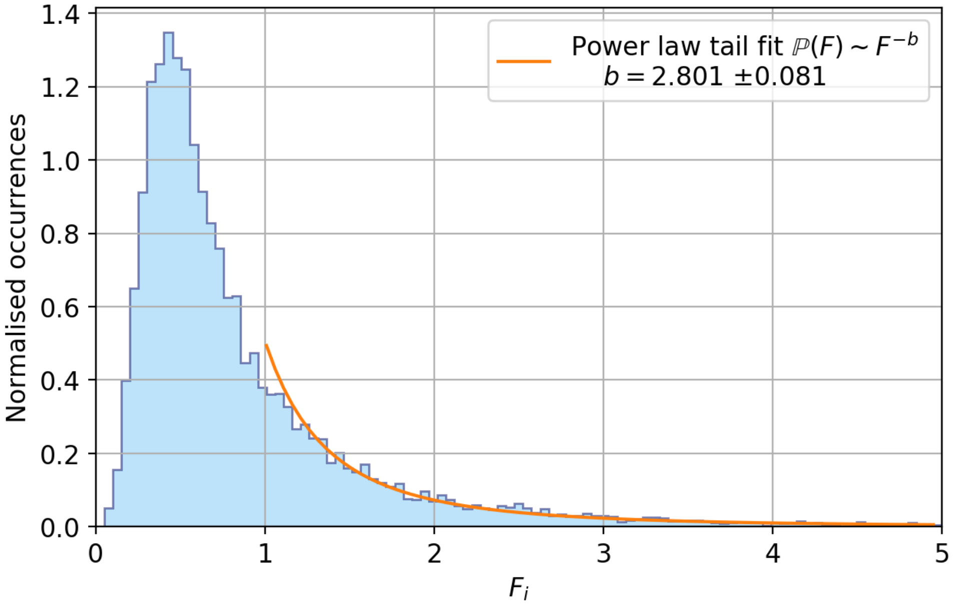

Figure 3.3.9: Self-consistent factor distribution for a Barabási-Albert network with 15000 nodes, $m = 4$. The histogram contains 100 bins from 0 to 5. The orange line is a power-law fit on the right tail.

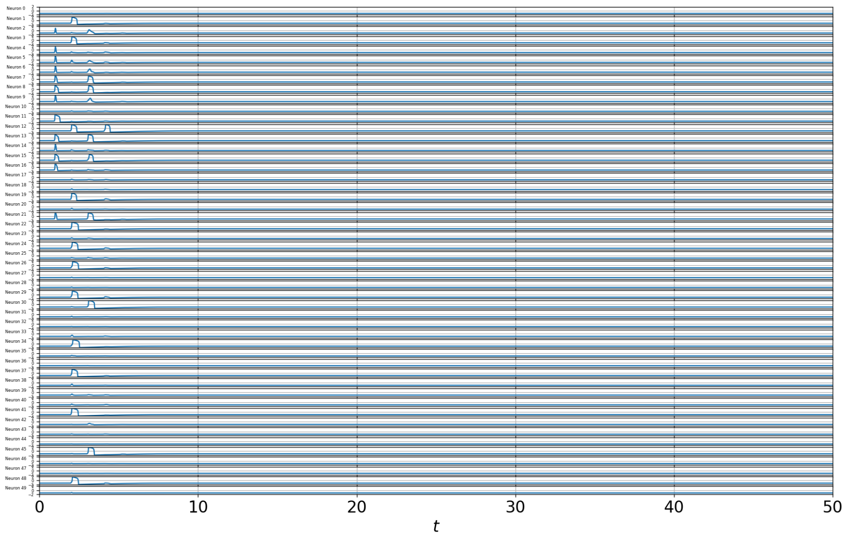

Figure 3.3.10: Activity plot for the Barabási-Albert network at $r = 0.5$.

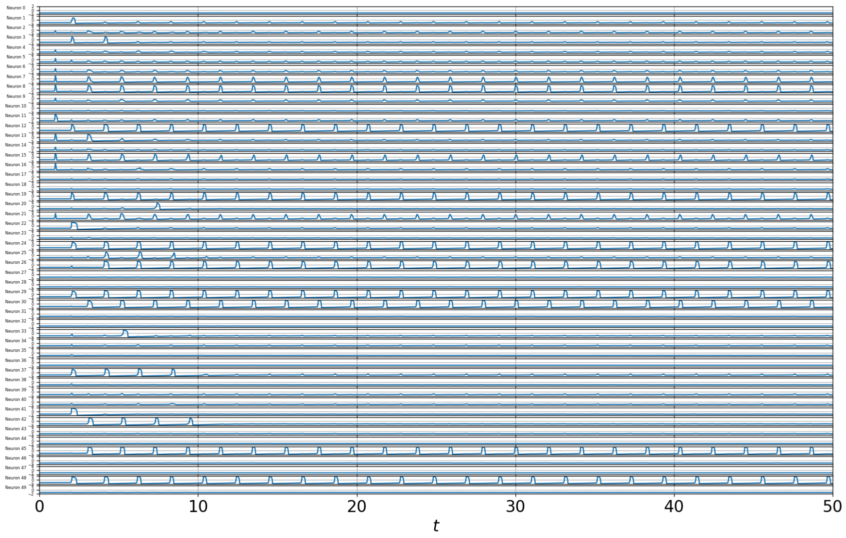

Figure 3.3.11: Activity plot for the Barabási-Albert network at $r = 1$.

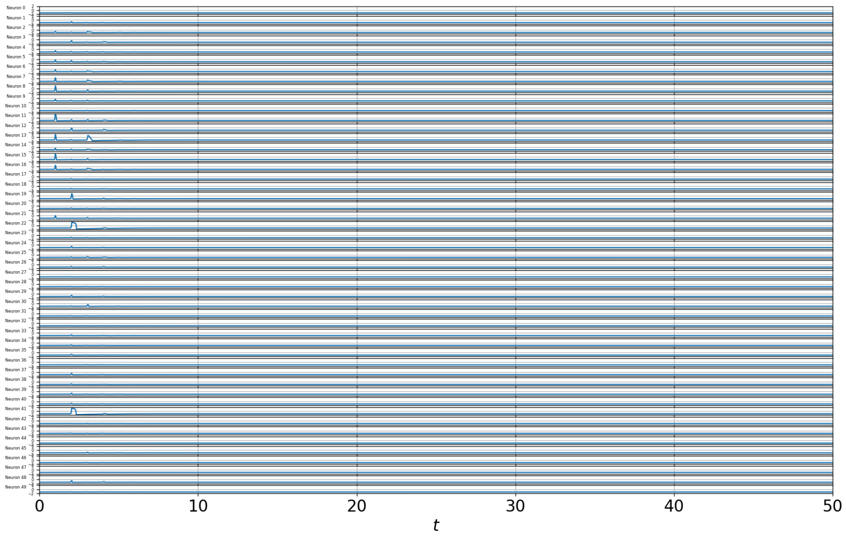

Figure 3.3.12: Activity plot for the Barabási-Albert network at $r = 1.5$.

3.4 Chapter summary and concluding remarks

We can summarise the results for this chapter as follows:

- Within the out-degree normalised scheme, it is not possible anymore to employ the simpler strategies used for the in-degree normalised scheme.

- To try and approach the problem of synchronisation, we work with a self consistent hypothesis, which, as discussed in Section 3.1, has some drawbacks, but allows for a first understanding of the synchronisation mechanisms that take place within this normalisation scheme.

- The relevant quantity in this approach is the self consistent factor F_i, which determines the propensity of node i to enter a possibly existing synchronous state. In particular, it is observed that the relevant set of nodes which must be stimulated in order to have activity on the network, even if not synchronous, is that whose members have an higher than average F_i.

- A Mean Field approximation can be employed, by hypothesising that each node connects to $\langle d \rangle$ other nodes, thus yielding a self consistent factor of 1 for each node.

- The quality of this approximation depends on the features of the statistical distribution of F_i, which can be obtained via a Monte Carlo approach.

- For the Erdős-Renyi model, the F_i distribution fits well a Gaussian shape, pointing out a Central Limit Theorem behaviour that can be verified by checking that the mean and variance of the inverse degree are finite, under the Erdős-Renyi hypotheses. To leading order, the distribution is approximated by
$$F_i \sim 1 + \frac{\mathrm{G}(0,1)}{\sqrt{Np}}$$
The Mean Field Approximation works well on this architecture, reasonably due to the distribution of F_i, even though some threshold-lowering effects can be present, as it is discussed in Section 3.3.

- On the Watts-Strogatz model, we observe a good behaviour of the Mean Field Approximation, similarly to what is observed for the Erdős-Renyi model, although with minor differences.

- As expected, for the Barabási-Albert model, the situation is largely different. The distribution of F_i has a power law tail, and is indeed peaked

> below the Mean Field value of 1. In this setting the Mean Field Approximation fails, and even at large value of the coupling strength, the network fails to synchronise, displaying only some scattered activity.

This second normalisation scheme presents several promising directions for future development. A first one is the application of this Mean Field Approach to more realistic networks, for example with a hyerarchical architecture. Another interesting feature to study would be to consider the *attractivity* of the stable states, so to understand how the attraction basins of synchronised and desynchronised phases behave for various parameter values. From a computational point of view, it would be very interesting to run larger size simulations, so to observe the properties of our approximation in relation to system scaling, and mitigate the effect of finite size. Finally, from a theoretical standpoint, it would be important to lay on more rigorous foundations this Mean Field Approach, and possibly link the study of neuronal synchronisation transitions to the theory of Phase Transitions and eventually to Out-of-equilibrium Statistical Mechanics.

Conclusion

In this dissertation we have analysed the phenomenon of neuronal synchronisation on complex network architectures in two different normalisation schemes.

In Chapter 2 we have dealt with the most studied case of in-degree normalised connections, for which a theory to explain the stability of the synchronous solution already exists, with the aim to understand the conditions under which it is possible to destabilise a completely synchronous solution. From previous results on the dynamics of such networks we have been able to assess that the only self-sustaining alternative to complete synchronisation is bipartite synchrony, a state where two network partitions fire in an alternating fashion. Fundamental facts from Spectral Graph Theory allow us to identify the network architectures on which such states can arise with bipartite networks.

In light of these findings we devise an algorithm that by iteratively removing links from a given network, reduces its structure to a bipartite network. The peculiarity of the algorithm is its usage of matrix perturbation theory, akin to the time independent perturbation theory of non-relativistic quantum mechanics, in finding out which link should be cut at each step, so to obtain a bipartite network. Moreover, we present a link between this approach and a previously introduced notion of Spectral Centrality. The algorithm is tested, and it is shown to be able to destabilise the synchronous state in the three most common random network architectures. The robustness of the mentioned network models with respect to the action of the algorithm is tested as well, finding out that only the Erdős-Renyi model appears to be able to partially retain its original structure, while the Watts-Strogatz and Barabási-Albert models appear to lose respectively all small-world and scale-free features in the process.

A continuation of this line of work can take different directions. Focusing on the dynamical aspects, multipartite states could be considered, working with directed networks. In such a context, the algorithm would have to be adapted so to deal with a complex spectrum. On the other hand, focusing on the network science and neuroscientific contents, it would be interesting to consider more realistic network architectures, possibly taken from data, so to drive the research towards data oriented modelling. Finally, from an algorithmic point of view, it would be important to try and isolate precisely the network features that our algorithm tends to destroy, so to understand how it could be improved. From a physiological point of view, the in-degree normalised scheme corresponds to neurons which assign the same relevance at each of their afferent synapses. This is not motivated in principle, and so we proceed to study a different scheme, where the output of neurons is normalised, so to reproduce schematically an effect of subdivision among the efferent postsynaptic neurons of a fixed amount of neurotransmitter.

In Chapter 3 we deal with the less studied case of out-degree normalised couplings. We put forward a self-consistent approach based on the role of a self-consistent factor F_i for each node, function of the adjacent edge weights, in changing the *effective* coupling strength felt by the neurons, and explore its behaviour and its shortcomings. By analysing with it the synchronisation of a small hand-built graph, we can state that it is reasonable to consider the thresholds it allows us to find as upper bounds on the minimal coupling strength for collective synchronisation to be able to take place.

To move towards larger complex network architectures we formulate a Mean Field Approximation, which approximates F_i with $\bar{F} = 1$, its average value. We expect this approach to perform better on networks with a high degree of regularity. Here as well we perform tests on the three main random network architectures. For the Erdős-Renyi model, we find that the distribution of F fits well a Gaussian centered on $\bar{F}$, thus pointing out a Central Limit Theorem behaviour, which can be traced back to the independence of the degrees of different nodes on an E-R network, and can be recovered analytically in a leading order approximation. In fact, simulations show a reasonable accord with the Mean Field Approximation, up to a general reduction of the thresholds which we hypothesize to be due to a transition from a mutual interaction to a driver-peripheral dynamics once large enough synchronised clusters emerge across the network and to small size effects. We find a similar accord with the Mean Field Approximation in the case of the Watts-Strogatz model, as we would expect due to the underlying lattice structure, albeit without finding a Gaussian distribution of the self-consistent factor. Conversely, on a Barabási-Albert network, we observe a very different scenario. The self-consistent factor distribution is indeed highly skewed, peaked well-below $\bar{F}$, and with a power-law decaying tail towards high values of F_i. Accordingly to these features, we observe that the Mean Field Approximation fails, as we are not able to observe any synchronisation, regardless of the coupling strengths considered.

The second approach presented in this dissertation is perhaps the most promising for future ampliation. A continuation would entail first a solid understanding of the proposed Mean Field Approximation, especially in terms of the scaling properties of the distribution of F_i for the main random network models. This aspect can be approached both theoretically and numerically, but a theoretical foundation would prove very useful, possibly linking the properties of F_i directly to well-defined properties of the network (e.g. small-world, scale-free), so to be able to provide useful results also in experimental settings, where the generating algorithm behind a given structure is not known. Another relevant contribution would surely come from the possible treatment of these models within a Statistical Physics approach, in terms of the analogy between Delayed Differential Equations and Ginzburg-Landau theories that has been explored, for example, for some traffic models.

Appendices

Appendix A

Physical interpretation of the Laplacian Matrix

Let us consider a diffusion process taking place on an undirected network, let Φ_i be a temperature, defined on each network site, and let heat diffuse from a site to the adjacent ones, according to the Newton law of cooling, the equations for this diffusion process are given by

$$\dot{\Phi}_i(t) = -k \sum_{j=1}^{N} A_{ij} \left[\Phi_i(t) - \Phi_j(t)\right] \tag{A.1}$$

with k being a constant proportional to the heat transfer coefficient.

$$\begin{aligned} \dot{\Phi}_i(t) &= -k \sum_j A_{ij} \left(\Phi_i(t) - \Phi_j(t)\right) \\ &= -k \sum_j \left(\delta_{ij} D_{ii} - A_{ij}\right) \Phi_j(t) \\ &= -k \sum_j L_{ij} \Phi_j(t) \end{aligned}$$

where L is the Laplacian matrix, as it has been defined in Equation (1.8). If we write down, for comparison, the heat equation

$$\frac{\partial \Phi(t, \mathbf{x})}{\partial t} = \nabla^2 \Phi(t, \mathbf{x}) \tag{A.2}$$

we realise immediately that the matrix $-L$ plays the role of the Laplacian on the network structure, hence the name of *Network Laplacian*.

Appendix B

Properties of stochastic matrices

We present here some useful facts about stochastic matrices for usage throughout this work. We follow the presentation by Gantmacher [Gan59]. We begin with the definition of stochastic matrix.

Definition B.0.1 (Stochastic matrix). Let M be an $n \times n$ matrix with non-negative real entries, $M_{ij} \geq 0\, \forall i,\ j$. Such a matrix is called *stochastic* if and only if

$$\sum_{j=1}^{n} M_{ij} = 1 \quad \forall i \in \{1, \dots, n\} \tag{B.1}$$

such a matrix can also be referred to as *row-stochastic*, so to stress that the unit-sum condition is on the rows. In the following we shall also consider the case where, under the same non-negativity constraints, such a condition is set on the columns, i.e.

$$\sum_{i=1}^{n} M_{ij} = 1 \quad \forall j \in \{1, \dots, n\} \tag{B.2}$$

which will be referred to as a *column-stochastic* matrix.

First we state a property of row stochastic matrices, that is widely used in Chapter 2, when the in-degree normalised scheme is studied.

Theorem B.0.1. Let M be an $n \times n$ matrix with non-negative entries. M is row-stochastic if and only if

$$M\mathbf{1}_n = \mathbf{1}_n \qquad \mathbf{1}_n = (\underbrace{1, 1, \dots, 1}_{n \text{ times}}) \tag{B.3}$$

so any matrix with non-negative entries is row-stochastic if and only if it accepts $\mathbf{1}_n$ as a right-eigenvector, with eigenvalue 1. The more relevant theorem, that allows us to discuss the stability in general terms whenever the connectivity matrix is stochastic, is the following.

Theorem B.0.2 (Spectral bound for stochastic matrices)**.** Let M be an $n \times n$ row-stochastic matrix. Let $\{m_1, m_2, \dots, m_n = 1\}$ be its spectrum. The following holds

$$|m_i| \leq 1 \quad \forall i \in \{1, \dots, n\} \tag{B.4}$$

where by $|\cdot|$ we denote the complex modulus.

this theorem states that the whole spectrum of a row stochastic matrix is contained within the unit circle in the complex plane. We use this fact in the present work, to study the stability of the completely synchronous state, and characterise the possible synchrony-breaking perturbations. Moreover, notice that this result remains true for column-stochastic matrices, due to the fact that for any square matrix, A and A^T share the same spectrum, even though the leading eigenvalue 1 will no longer be associated to eigenvector $\mathbf{1}_n$.

Appendix C

The `RADAR 5` numerical integration algorithm

All numerical simulations involving Delayed Differential equations, in the present work, have been performed using the `RADAR 5` integration algorithm [GH05], by Nicola Guglielmi and Ernst Hairer. The algorithm is designed for the solution of initial value problems of the form

$$\begin{aligned} M\dot{\mathbf{y}}(t) &= \mathbf{f}\left(t, \mathbf{y}(t), \mathbf{y}(\alpha_1(t, \mathbf{y}(t)), \ldots, \mathbf{y}(\alpha_m(t, \mathbf{y}(t)))\right) \\ \mathbf{y}(t_0) &= \mathbf{y}_0, \quad \mathbf{y}(t) = \mathbf{g}(t) \text{ for } t < t_0 \end{aligned} \tag{C.1}$$

where the real state variables $\mathbf{y}$ are arranged in a d dimensional vector, M is a $d \times d$ dimensional real matrix, and the $\alpha_i(t, \mathbf{y}(t)) \leq t$ are possibly time and state dependent delayed times. Many different kinds of problems can be cast in this form. For example, the matrix M could arise from the discretisation of a time-delayed Partial Differential Equation. The algorithm, in particular, does not need to invert M, therefore avoiding the disruption of its sparsity pattern, which can lead to problems due to small denominators or in general increase the numerical overhead of the computation, in the case, for example, of very large systems. M can also be singular, thus making the algorithm able to solve also Differential-Algebraic problems. A very relevant class of equations, solvable by the algorithm, is that of *singularly perturbed problems*, i.e. those problems where the mass matrix takes the form

$$M = \begin{pmatrix} I & 0 \\ 0 & \varepsilon I \end{pmatrix} \tag{C.2}$$

with $\varepsilon \ll 1$, which is clearly the case for the FitzHugh-Nagumo system. Moreover, these equations form an important class of so called *stiff* problems.

It is hard to give a precise and general definition of a stiff problem. In general, a problem is referred to as stiff, if one or more algorithms yield numerically unstable solutions to them, unless a very small integration step is set. Singularly perturbed problems, in particular, are stiff with respect to *explicit* integration algorithms. Numerical integration schemes can be cathegorised as explicit or implicit, depending on how the calculation of the future state of the system, most commonly at the next discrete integration time-step, is carried out:

- In an *explicit scheme*, the state at a future time is calculated via a function of the present state, and possibly of the system's history, when DDEs are considered. Schematically, and for a single delay τ

  $$\mathbf{y}(t + \Delta t) = \mathbf{F}(\mathbf{y}(t), \mathbf{y}(t - \tau)) \tag{C.3}$$

  The construction and choice of the form of $\mathbf{F}$ is an integral part of the algorithm design process. In principle, these methods are not very computationally expensive, as a step requires a single functional evaluation.

- In an *implicit scheme*, conversely, the state at a future time is found as the solution to a, generally nonlinear, equation. For the same hypotheses of the previous points, we will schematically have

  $$G(\mathbf{y}(t + \Delta t), \mathbf{y}(t), \mathbf{y}(t - \tau)) = 0 \tag{C.4}$$

  As in the previous case, the choice and determination of G is a fundamental part of the algorithm design. A step of these types of algorithm is more costly, in principle, as it requires the solution of equation (C.4), which in general entails an iterative process in itself. Moreover, these schemes are generally more complicated to implement.

Despite the general aspects just presented, the convenience and practicality of one or the other type of scheme depends on the problem one is solving. When one is confronting stiff problems, for example, implicit algorithms are more useful than explicit ones, as the overhead given by the solution of the nonlinear equation can often be computationally cheaper than the abnormally small integration step that an explicit method would require.

`RADAR 5` is an implicit scheme, making it suited to the solution of stiff DDE problems, as those regularly encountered in the writing of the present work. The algorithm itself is based on a 3 stage RADAU-II collocation method [WH96]. The original codebase is written in `FORTRAN 90`, allowing for a highly optimised performance also on large systems. In the present work, we have accessed the integration routines via a Python binding available on GitHub, which has partially been modified by the author of this dissertation so to allow the simulation of large systems.

Appendix D

Pseudocode for the Bipartition algorithm

We present in this appendix the pseudocode, Algorithm 1, descring the bipartition algorithm. The actual implementation has been performed in the Python language, making use of the main modules for scientific computing such as `numpy` and `scipy`. The management of network objects has been performed using the `networkx` module. In particular, the connectedness check has been performed using the `networkx.is_connected` method, which implements a Breadth First Search [Lee61]. Its asymptotic order is of $\mathcal{O}(V+E)$, where E is the number of edges, and V the number of nodes within the graph.

An alternative way of performing this check is via the algebraic connectivity, or Fiedler value λ_2, the second Laplacian eigenvalue, which collapses to zero if and only if the graph is disconnected. In principle, we could have evaluated the effect of the link removal on the Fiedler value via an analogous perturbative expression to that used for λ_N, avoiding any moves for which the algebraic connectivity would collapse to 0. Indeed, if we look at the algorithm as to a *gradient descent*, we understand that the bipartition implies the minimisation of a functional of the normalised Laplacian

$$\mathcal{L}(L_{RW}) = \frac{1}{2}\left(\lambda_N(L_{RW}) - 2\right)^2$$

where the descent steps are performed discretely on the graph structure, by removing links. In this context, the requirement that $\lambda_2 > 0$, becomes a unilateral constraint, which would entail the modification of the aforementioned functional. The form of the functional, via its gradient, dictates the weight attributed to the variation of each of the two eigenvalues, when determining the direction of steepest descent. During development, this balancing process has proved to be challenging, the behaviour of the algorithm changing quite abruptly for slight changes in the functional. For these reasons we chose to adopt a combinatorial solution, restrictedly to the connectedness check.

Algorithm 1: Bipartition algorithm

```
Result: Bipartite graph $G_b(V, L_b)$
Input: Initial graph $G(V, L)$, tolerance $tol = 10^{-6}$
begin
    Evaluate initial $\lambda_N$ for graph $G$;
    Initialise working edge list $L_w = L$;
    while $|\lambda_N - 2| > tol$ do
        Initialise empty list $l_c$;
        /* It will store links candidate for removal.     */
        for $(l, m)$ in $L_w$ do
            Evaluate $\dot{\lambda}_N(0)$ for $(l, m)$ using (2.37);
            if $\dot{\lambda}_N(0) > 0$ then
                Append $(l, m)$ to $l_c$;
            end
        end
        if $l_c$ is empty then
            return $-1$;
            /* No candidate link removal can improve $\lambda_N$,
               exit with error.                               */
        else
            Sort $l_c$ according to descending $\dot{\lambda}_N(0)$ values
        end
        for $(l, m)$ in $l_c$ do
            Build subgraph $G_w$ from links in $L_w \setminus (l, m)$;
            if $G_w$ is connected then
                Set $L_w \setminus (l, m)$ as new $L_w$;
                break;
            else
                if $(l, m)$ is the last element in $L_w$ then
                    return $-1$ ;
                    /* The removal of any candidate would
                       disconnect $G$, exit with error.       */
                end
            end
        end
    end
    Build $G_b$ from the links in $L_w$;
    Return $G_b$;
end
```

Appendix E

Estimates on the distribution of F_i

We set out to show that under the hypotheses of Section 3.3, F_i is approximately distributed according to a Gaussian distribution with mean 1 and standard deviation scaling as $\sim \frac{1}{\sqrt{N}}$ with respect to the number of nodes. First we need to estimate the mean and variance of $\frac{1}{d}$, when d is distributed according to the zero-truncated Poisson distribution with parameter μ.

We can approximate the mean by introducing $\Delta d = d - \mu$ and taking the leading term in a binomial series expansion

$$\begin{aligned}\left\langle \frac{1}{d} \right\rangle_c &= \frac{1}{e^{\mu}-1} \sum_{d=1}^{\infty} \frac{\mu^d}{d!} \frac{1}{d} \\ &= \sum_{d=1}^{\infty} \frac{\mu^d}{d!} \frac{1}{\mu + \Delta d} \simeq \sum_{d=1}^{\infty} \frac{\mu^d}{d!} \frac{1}{\mu} \left(1 - \frac{\Delta d}{\mu}\right) \\ &= \frac{1}{\mu}\end{aligned} \tag{E.1}$$

And with the same procedure we can approximate the second inverse absolute moment as

$$\left\langle \frac{1}{d^2} \right\rangle_c \simeq \frac{1}{\mu^2} + \frac{3}{\mu^3} \tag{E.2}$$

so that in turn the variance is approximated by

$$\mathrm{Var}\left[\frac{1}{d}\right] \simeq 3\mu^{-3} \tag{E.3}$$

Using these estimates and considering a large enough d_i, the sum in the definition of F_i, by virtue of the Central Limit Theorem, can be approximated by

$$F_i \simeq \frac{d_i}{\mu} + \sqrt{\frac{3d_i}{\mu^3}} \mathrm{G}(0,1) \tag{E.4}$$

where by $\mathrm{G}(0,1)$ we denote the standard Gaussian distribution with mean 0 and unit variance. This expression gives an approximation for the fluctuations of the self-consistent factor of node i caused by the degrees of its neighbours when the degree of i is fixed, and most importantly tells us how they scale with respect to μ. The degree d_i, although, is not fixed, and has fluctuations of its own, which we now take into account. Recalling that d_i is distributed according to a Poisson distribution with parameter μ, we can average both over d_i and the Gaussian fluctuations in Expression (E.4) and recover the Mean Field Value $\bar{F} = 1$

$$\langle F_i \rangle \simeq \frac{\langle d_i \rangle}{\mu} + \left\langle \sqrt{\frac{3d_i}{\mu^3}} \right\rangle \langle \mathrm{G}(0,1) \rangle = 1 \tag{E.5}$$

where we have factorised the expectation in the second term owing to the fact that d_i and $\mathrm{G}(0,1)$ are independent random variables. Now we move on to estimating the variance of F_i, so to understand which fluctuations are dominant, and how they do scale with μ. First we approximate the term $\propto \sqrt{d_i}$ in a binomial series expansion centered in μ, obtaining Expression (E.6).

$$F_i \simeq \frac{d_i}{\mu} + \frac{\sqrt{3}}{\mu}\left(1 + \frac{\Delta d_i}{2\mu}\right)\mathrm{G}(0,1) \tag{E.6}$$

Now, reverting the definition $\Delta d_i = d_i - \mu$, we are able to obtain Expression (E.7)for F_i

$$F_i \simeq \frac{d_i}{\mu} + \frac{\sqrt{3}}{2\mu^2} d_i \mathrm{G}(0,1) - \frac{\sqrt{3}}{2\mu}\mathrm{G}(0,1) \tag{E.7}$$

which we can use to calculate an approximation for the variance of F_i.

Indeed, we recall that for X, Y independent random variables the following identity holds

$$\mathrm{Var}\left[XY\right] = \mathbb{E}\left[X^2\right]\mathbb{E}\left[Y^2\right] - \mathbb{E}\left[X\right]^2\mathbb{E}\left[Y\right]^2$$

which we use for the second term in the sum on the r.h.s. of Equation (E.7), obtaining

$$\mathrm{Var}\left[F_i\right] \simeq \frac{1}{\mu} + \frac{3}{2\mu^2} + \frac{3}{4\mu^3} \tag{E.8}$$

Equation (E.8) contains two important pieces of information. The first is that the leading order in μ of the approximate variance is $\propto \frac{1}{\mu}$, and the second is that this term stems from the first term in the r.h.s. of Equation (E.6), which allows us to trace back the dominant fluctuations to the Poissonian fluctuations in the degree of node i, and not in those of its neighbours. Furthermore, since we are considering large values of μ, we expect the shape of the distribution of

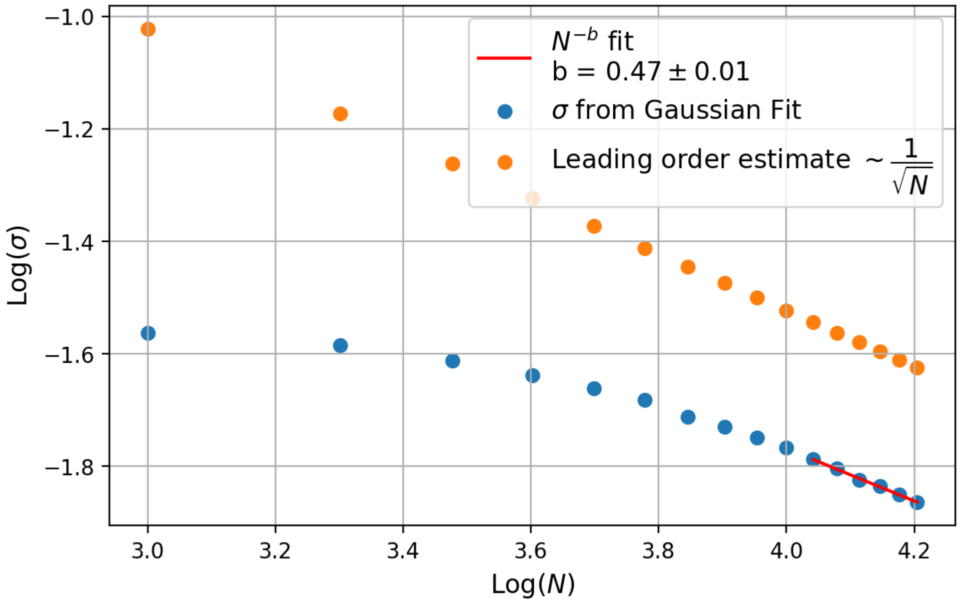

Figure E.0.1: Standard deviation values obtained from fitting on simulated Erdős-Renyi networks compared with the estimate $\propto N^{-1/2}$. A power law regression is shown in red, performed on the six largest values of N. N value vary from 1000 to 16000.

F_i to be well approximated by a Gaussian, as it is expected of Poisson variables in the limit of large parameter. Recalling the definition of $\mu = Np$ we find that at leading order, the standard deviation of F_i scales as indicated in the following Equation (E.9).

$$\mathrm{SD}\left[F_i\right] \propto \frac{1}{\sqrt{N}} \tag{E.9}$$

In conclusion, the findings just discussed allow us to state that for large N and large enough μ, the distribution of the self-consistent factor F_i will be well approximated by a Gaussian centered in $\bar{F} = 1$, with a variance scaling as $\propto \frac{1}{\mu} = \frac{1}{Np}$.

$$F_i \sim 1 + \frac{\mathrm{G}(0,1)}{\sqrt{Np}}$$

In Figure E.0.1 we show the results of a numerical experiment carried out to compare the scaling of the standard deviation of F_i on simulated Erdős-Renyi networks with our estimate. For large values of N, the power law regression produces a value of -0.47 ± 0.01, which is close to the estimate, although not strictly compatible. Due to the asymptotic nature of the approximation, better results could probably be obtained by considering larger networks, a possibility that in the present work has been limited by the available computing power.

Bibliography

[AW30] A. Andronow and A. Witt. "Zur Theorie des Mitnehmens von Van der Pol". In: *Archiv für Elektrotechnik* 24.1 (1930), pp. 99–110.

[BA99] Albert-László Barabási and Réka Albert. "Emergence of Scaling in Random Networks". In: *Science* 286.5439 (1999), pp. 509–512.

[BBH16] Christopher S von Bartheld, Jami Bahney, and Suzana Herculano-Houzel. "The search for true numbers of neurons and glial cells in the human brain: a review of 150 years of cell counting". In: *Journal of Comparative Neurology* 524.18 (2016), pp. 3865–3895.

[BZG18] Danielle Bassett, Perry Zurn, and Joshua Gold. "On the nature and use of models in network neuroscience". In: *Nature Reviews Neuroscience* 19 (July 2018).

[Chu97] Fan R.K. Chung. *Spectral graph theory*. 92. American Mathematical Soc., 1997.

[Del+20] Fabio Della Rossa et al. "A network model of Italy shows that intermittent regional strategies can alleviate the COVID-19 epidemic". In: *Nature communications* 11.1 (2020), pp. 1–9.

[ER59] Paul Erdős and Alfred Renyi. "On Random Graphs I". In: *Publicationes Mathematicae Debrecen* 6 (1959), pp. 290–297.

[Fit61] Richard FitzHugh. "Impulses and physiological states in theoretical models of nerve membrane". In: *Biophysical journal* 1.6 (1961), pp. 445–466.

[Gan59] Felix Ruvimovich Gantmacher. *The Theory of Matrices*. AMS Chelsea Publishing, 1959.

[Ger+20] Moritz Gerster et al. "FitzHugh–Nagumo oscillators on complex networks mimic epileptic-seizure-related synchronization phenomena". In: *Chaos: An Interdisciplinary Journal of Nonlinear Science* 30.12 (2020), p. 123130.

[GH05] Nicola Guglielmi and Ernst Hairer. "Users' guide for the code RADAR5-version 2.1". In: *Rapport technique, Università dell'Aquila, Italy* (2005).

[GSC00] Luc J. Gentet, Greg J. Stuart, and John D. Clements. "Direct Measurement of Specific Membrane Capacitance in Neurons". In: *Biophysical Journal* 79.1 (2000), pp. 314–320.

[He+19] Zhiwei He et al. "Perturbation analysis and comparison of network synchronization methods". In: *Physical Review E* 99.5 (2019), p. 052207.

[HH52] Alan L. Hodgkin and Andrew F. Huxley. "A quantitative description of membrane current and its application to conduction and excitation in nerve". In: *The Journal of physiology* 117.4 (1952), pp. 500–544.

[HSS08] Aric A. Hagberg, Daniel A. Schult, and Pieter J. Swart. "Exploring Network Structure, Dynamics, and Function using NetworkX". In: *Proceedings of the 7th Python in Science Conference*. Ed. by Gaël Varoquaux, Travis Vaught, and Jarrod Millman. Pasadena, CA USA, 2008, pp. 11–15.

[Huy99] Christiaan Huygens. *Oeuvres complètes de Christiaan Huygens*. Vol. 8. M. Nijhoff, 1899.

[Izh06] Eugene M. Izhikevich. *Dynamical Systems in Neuroscience: The Geometry of Excitability and Bursting*. The MIT Press, July 2006. ISBN: 9780262276078.

[Kan+11] I. Kanter et al. "Nonlocal mechanism for cluster synchronization in neural circuits". In: *EPL (Europhysics Letters)* 93.6 (2011), p. 66001.

[KK89] Tomihisa Kamada and Satoru Kawai. "An algorithm for drawing general undirected graphs". In: *Information Processing Letters* 31.1 (1989), pp. 7–15.

[Kur75] Yoshiki Kuramoto. "Self-entrainment of a population of coupled non-linear oscillators". In: *International Symposium on Mathematical Problems in Theoretical Physics*. Ed. by Huzihiro Araki. Berlin, Heidelberg: Springer Berlin Heidelberg, 1975, pp. 420–422.

[Lax07] Peter D. Lax. *Linear Algebra and Its Applications*. Pure and Applied Mathematics: A Wiley Series of Texts, Monographs and Tracts. Wiley, 2007.

[Lee61] C. Y. Lee. "An Algorithm for Path Connections and Its Applications". In: *IRE Transactions on Electronic Computers* EC-10.3 (1961), pp. 346–365.

[Leh+11] J. Lehnert et al. "Loss of synchronization in complex neuronal networks with delay". In: *EPL (Europhysics Letters)* 96.6 (Mar. 2011), p. 60013.

[Leh+14] Klaus Lehnertz et al. "Evolving networks in the human epileptic brain". In: *Physica D: Nonlinear Phenomena* 267 (2014). Evolving Dynamical Networks, pp. 7–15.

[Leh16] Judith Lehnert. *Controlling Synchronization Patterns in Complex Networks*. Springer Verlag, 2016. ISBN: 9783319251158.

[NAY62] Jinichi Nagumo, Suguru Arimoto, and Shuji Yoshizawa. "An active pulse transmission line simulating nerve axon". In: *Proceedings of the IRE* 50.10 (1962), pp. 2061–2070.

[New18] Mark Newman. *Networks*. Oxford university press, 2018.

[PC98] L. M. Pecora and T. L. Carroll. "Master Stability Functions for Synchronized Coupled Systems". In: *Phys. Rev. Lett.* 80 (10 Mar. 1998), pp. 2109–2112.

[PR12] Scott D. Pauls and Daniel Remondini. "Measures of centrality based on the spectrum of the Laplacian". In: *Physical Review E* 85.6 (2012), p. 066127.

[PRK01] Arkady Pikovsky, Michael Rosenblum, and Jürgen Kurths. *Synchronization: A Universal Concept in Nonlinear Sciences*. Cambridge Nonlinear Science Series. Cambridge University Press, 2001.

[Ram+19] Lukas Ramlow et al. "Partial synchronization in empirical brain networks as a model for unihemispheric sleep". In: (Apr. 2019).

[Rot13] Jason S. Rothman. "Modeling Synapses". In: *Encyclopedia of Computational Neuroscience*. Ed. by Dieter Jaeger and Ranu Jung. New York, NY: Springer New York, 2013, pp. 1–15.

[Sin99] W. Singer. "Neuronal synchrony: a versatile code for the definition of relations?" In: *Neuron* 24.1 (Sept. 1999), pp. 49–65, 111–25. ISSN: 0896-6273.

[Sto+02] Lewi Stone et al. "Complex Synchronization Phenomena in Ecological Systems". In: *AIP Conference Proceedings* 622.1 (2002), pp. 476–488.

[Tur98] Giorgio Turchetti. *Dinamica classica dei sistemi fisici*. Zanichelli, 1998.

[VV27] Balthasar Van der Pol and Jan Van Der Mark. “Frequency demultiplication”. In: *Nature* 120.3019 (1927), pp. 363–364.

[Wan+08] Samuel S-H Wang et al. “Functional trade-offs in white matter axonal scaling”. In: *Journal of neuroscience* 28.15 (2008), pp. 4047–4056.

[WH96] Gerhard Wanner and Ernst Hairer. *Solving ordinary differential equations II*. Vol. 375. Springer Berlin Heidelberg, 1996.

[Win67] Arthur T. Winfree. “Biological rhythms and the behavior of populations of coupled oscillators”. In: *Journal of theoretical biology* 16.1 (1967), pp. 15–42.

[WS98] Duncan J. Watts and Steven H. Strogatz. “Collective dynamics of ‘small-world’networks”. In: *nature* 393.6684 (1998), pp. 440–442.