
# FROM COMMUNITY STRUCTURE TO THE PHYSICS OF MULTIPLEX NETWORKS

**Clara Granell Martorell**

UNIVERSITAT ROVIRA I VIRGILI
FROM COMMUNITY STRUCTURE TO THE PHYSICS OF MULTIPLEX NETWORKS
Clara Granell Martorell

UNIVERSITAT ROVIRA I VIRGILI
FROM COMMUNITY STRUCTURE TO THE PHYSICS OF MULTIPLEX NETWORKS
Clara Granell Martorell

Clara Granell Martorell

# FROM COMMUNITY STRUCTURE TO THE PHYSICS OF MULTIPLEX NETWORKS

PhD Thesis
Supervised by Dr. Alex Arenas and Dr. Sergio Gómez

Departament d'Enginyeria Informàtica i Matemàtiques
November 2015

UNIVERSITAT
ROVIRA i VIRGILI

UNIVERSITAT ROVIRA I VIRGILI
FROM COMMUNITY STRUCTURE TO THE PHYSICS OF MULTIPLEX NETWORKS
Clara Granell Martorell

UNIVERSITAT ROVIRA i VIRGILI

**Departament d'Enginyeria Informàtica i Matemàtiques**
Avinguda Països Catalans, 26
43007, Tarragona.

I STATE that the present study, entitled FROM COMMUNITY STRUCTURE TO THE PHYSICS OF MULTIPLEX NETWORKS, presented by CLARA GRANELL MARTORELL for the award of the degree of Doctor, has been carried out under our supervision at the Department DEPARTAMENT D'ENGINYERIA INFORMÀTICA I MATEMÀTIQUES of this university.

Tarragona, $15^{th}$ November 2015.

The supervisors,

Dr. Alex Arenas Moreno

Dr. Sergio Gómez Jiménez

UNIVERSITAT ROVIRA I VIRGILI
FROM COMMUNITY STRUCTURE TO THE PHYSICS OF MULTIPLEX NETWORKS
Clara Granell Martorell

*To my grandmother, Josefina Sardà, a strong and modern woman who has always been supportive of every decision I've made.*

***

*Per la meva àvia, Josefina Sardà, una dona forta i moderna que sempre m'ha fet tant de costat en tot allò que he decidit fer.*

UNIVERSITAT ROVIRA I VIRGILI
FROM COMMUNITY STRUCTURE TO THE PHYSICS OF MULTIPLEX NETWORKS
Clara Granell Martorell

# Acknowledgements

I would like to thank many people who have helped me through the completion of this thesis. I will be always thankful for the guidance, help and endless discussion hours that my two supervisors Alex Arenas and Sergio Gómez have shared with me. They already know that, but it's worth telling them again, that they have been a family to me. They've taught me how to do research, they have listened all my conference presentation rehearsals, and they have been supportive even when I was too stressed out to be a nice company. I know my experience is not common, they really do things differently, and I just hope that sometime I can follow their path and be as good supervisor as they were to me.

I would also like to thank my family, who has been very supportive and has helped me to walk this long journey without feeling alone. I'd like to specially thank my mother for her unconditional love and support; my dad for always demanding the best of me, and my little brother Joan for being always there when I needed to go out and get some pizza. I'd like to have a special mention for my grandmother, the best granny I could have had, a strong woman that has always been my inspiration, and who has been supportive of all my academic efforts since I started going to school. Also thanks to all my aunts and uncles and cousins, for always being so happy of my success as if it was theirs.

I need also to thank my good friends Cristina and Mireia, for helping me keep my sanity and taking me out sometimes, and listening to my endless speeches about how hard it is to do a PhD. I'd like to acknowledge also my lab mates, who have been much more than just mere coworkers. With them I've shared a countless amount of hours discussing about science, having coffee and hanging out, special thanks to Elisa, Albert, Manlio, Serafina, Joan, Hongrun, Sebas and Pau.

I'd also like to acknowledge the James S. McDonnell Foundation for awarding me one of their Postdoctoral Fellowships and sending me the notification in the middle of the writing of this document, providing me with a much desired peace of mind that has let me finish this document happier than ever.

UNIVERSITAT ROVIRA I VIRGILI
FROM COMMUNITY STRUCTURE TO THE PHYSICS OF MULTIPLEX NETWORKS
Clara Granell Martorell

# CONTENTS

Contents

UNIVERSITAT ROVIRA I VIRGILI
FROM COMMUNITY STRUCTURE TO THE PHYSICS OF MULTIPLEX NETWORKS
Clara Granell Martorell

# LIST OF FIGURES

List of Figures

List of Figures

# LIST OF TABLES

UNIVERSITAT ROVIRA I VIRGILI
FROM COMMUNITY STRUCTURE TO THE PHYSICS OF MULTIPLEX NETWORKS
Clara Granell Martorell

# PREFACE

This document comprises the research work that I have carried during the years of my PhD, under the supervision of Professors Alex Arenas and Sergio Gómez, at the Universitat Rovira i Virgili, in Tarragona. The original works presented here are devoted to a better understanding of the structure and dynamics of complex networks, and particularly, of multiplex networks. I have been involved in the design, modeling, coding and writing of all of the works presented. Most of them were carried out by my supervisors and me exclusively, except for some collaborative works, in which I have to thank Dr. Sara Teller, Prof. Jordi Soriano, Dr. Richard K. Darst and Prof. Santo Fortunato for their expertise, help, and countless hours of scientific discussion that we spent to carry out some of the works that I present in this document.

The structure of this document is as follows: Chapter 1 is devoted to the introduction of the main concepts needed for a proper understanding of the works that I present subsequently. This introduction is divided in a brief introduction to complex networks (Sec. 1.1), an introduction to the analysis of the mesoscopic structure of networks (Sec. 1.2), the preliminaries of multilayer networks (Sec. 1.3) and the essentials of epidemic spreading (Sec. 1.4).

The rest of the document is devoted to present the contributions that I have made during these years and that have been published in academic journals. Chapter 2 comprises the work aimed to discover the mesoscopic structure of networks, either by extending existing methods, designing new approaches or creating benchmarks for such purpose. Chapter 3 is devoted to report works done concerning dynamical processes on top of multiplex networks, and Chapter 4 presents an experimental work aimed at characterizing, using the complex networks toolset, the functional structure of cultured neurons. Finally, in Chapter 5, conclusions and future perspectives of these works are offered.

This document was written between August and November 2015, and I have made big effort to ensure a fluid readability while respecting the details of the scientific works that I present. I hope you enjoy its reading.

*Clara Granell Martorell, Tarragona, 15$^{th}$ November 2015.*

UNIVERSITAT ROVIRA I VIRGILI
FROM COMMUNITY STRUCTURE TO THE PHYSICS OF MULTIPLEX NETWORKS
Clara Granell Martorell

# LIST OF PUBLICATIONS

Publications covered in this document:

- MESOSCOPIC ANALYSIS OF NETWORKS: APPLICATIONS TO EXPLORATORY ANALYSIS AND DATA CLUSTERING, by *Clara Granell, Sergio Gómez and Alex Arenas.* Chaos: An Interdisciplinary Journal of Nonlinear Science 21 (1), 016102 (2011).

  (Sections 2.1 and 2.2)

- UNSUPERVISED CLUSTERING ANALYSIS: A MULTISCALE COMPLEX NETWORKS APPROACH, by *Clara Granell, Sergio Gómez and Alex Arenas.* International Journal of Bifurcation and Chaos 22 (07), 1230023 (2012).

  (Sections 2.1 and 2.2)

- HIERARCHICAL MULTIRESOLUTION METHOD TO OVERCOME THE RESOLUTION LIMIT IN COMPLEX NETWORKS, by *Clara Granell, Sergio Gómez and Alex Arenas.* International Journal of Bifurcation and Chaos 22 (07), 1250171 (2012).

  (Section 2.3)

- DYNAMICAL INTERPLAY BETWEEN AWARENESS AND EPIDEMIC SPREADING IN MULTIPLEX NETWORKS, by *Clara Granell, Sergio Gómez and Alex Arenas.* Physical Review Letters 111 (12), 128701 (2013).

  (Section 3.1)

- COMPETING SPREADING PROCESSES ON MULTIPLEX NETWORKS: AWARENESS AND EPIDEMICS, by *Clara Granell, Sergio Gómez and Alex Arenas.* Physical Review E 90 (1), 012808 (2014).

  (Section 3.2)

- EMERGENCE OF ASSORTATIVE MIXING BETWEEN CLUSTERS OF CULTURED NEURONS, by *Sara Teller, Clara Granell, Manlio De Domenico, Jordi Soriano, Sergio Gómez and Alex Arenas.* PLOS Computational Biology 10(9): e1003796 (2014).

  (Section 4.1)

LIST OF PUBLICATIONS

- A BENCHMARK MODEL TO ASSESS COMMUNITY STRUCTURE IN EVOLVING NETWORKS, by *Clara Granell, Richard K. Darst, Alex Arenas, Santo Fortunato and Sergio Gómez.* Physical Review E 92, 012805 (2015).
(Section 2.4)

Other publications:

- STRUCTURAL PATTERNS IN COMPLEX SYSTEMS USING MULTIDENDROGRAMS, by *Sergio Gómez, Albert Fernández, Clara Granell and Alex Arenas.* Entropy 15 5464-5474 (2013)
- INFORMATION TRANSFER IN COMMUNITY STRUCTURED MULTIPLEX NETWORKS, by *Albert Solé-Ribalta, Clara Granell, Sergio Gómez and Alex Arenas.* Frontiers in Physics, 3 (2015)
- AUTOMATIC SLICING OF TEMPORAL DATA, by *Richard K. Darst, Clara Granell, Alex Arenas, Sergio Gómez, Jari Saramäki and Santo Fortunato. (In preparation).*

# 1

# INTRODUCTION

In its most basic definition, a network is a set of vertices and edges, sometimes referred to as nodes and links. There are many biological, technological and social systems that consist of individual components linked between them, such as computer networks, friendship relations between people or networks of interactions between proteins, to mention a few. This kind of systems is naturally suited to be represented as networks. This representation is particularly convenient when the object of study is the pattern of connections between elements, and not the individual characteristics of the single components. For instance, learning everything possible of water molecules still does not give the scientist an idea of why a collection of them will be in liquid state at 1°C and at solid state at -1°C. Instead, the answer for this kind of transition lies in a transformation in the organization of the network of their interactions. This approach holds true for a variety of different problems. Similarly, understanding the propagation of ideas between people would require to neglect the complexity within each human being composing the network and focus the analysis on the relations between people instead. In brief, representing a complex system as a network supposes a simplification of the system, stripping away all the details and reducing it to its bare bones, the patterns of interactions. This reduction in complexity of course implies loss of detail of the particular system, but it has a major advantage: it provides the scientist with a structure that is possible to handle mathematically.

## 1.1 INTRODUCTION TO COMPLEX NETWORKS

Complex networks science is an interdisciplinary field which borrows knowledge and techniques from graph theory, artificial intelligence and statistical physics. The term complex network was coined during the analysis of the power grid elec-

tricity distribution network and the Internet, to differentiate these networks from regular lattices, and additionally, to differentiate its study from the conventional mathematical techniques used in graph theory. Long before the complex network field was properly an established field, in the fifties, social scientists studied networks for representing the relations between humans in what they called social networks [215], although the analysis techniques still were relatively scarce. Graph theory was mathematically grounded during the sixties [29, 68, 69] and provided the qualitative change needed for the accurate analysis of networks. Forty years later, at the beginning of this century, the physicist's community became specially interested in this field and started using the common tools of statistical physics for their study [4, 25, 155, 217]. Up to date, the study of networks has continued growing, feeding from contributions of scientists in different fields, united in a single purpose: unraveling the delicate and intricate organization of networks of bewildering complexity.

Until the present day, a lot of network-like data has been collected and been the subject of study of a lot of academics. A very classic example of such networks is the collaboration network of Paul Erdős, depicted in Fig. 1.1. Paul Erdős was a very important Hungarian mathematician, who published circa 1400 papers in his lifetime, and made a very large number of collaborations by traveling around the world and working with the colleagues he visited. In the network, nodes account for the different scientists and edges connect pairs who have jointly authored a paper. This collaboration network also accounts as our first example of a social network, because, even though the relation depicted in this network is professional, it is reasonable to assume that two authors that have written a paper together know each other. Note that two mathematicians might have been friends but never published a paper together, relationship that is not captured in this particular example.

A different kind of network is the one that represents the Benguela marine ecosystem, which runs along southwestern Africa and the coastline of Angola, Namibia and South Africa, see Fig. 1.2. In this setup, nodes represent the species of animals that live within the area of study, and links are the trophic relations between a predator and its preys. Topologically speaking, this graph is different in nature than the previous in the sense that the interactions are directed: arrows point from a prey species to each one of his predators.

Deeper in the realm of biology, our third example is a metabolic network, a set of metabolic and physical processes that determine the physiological and

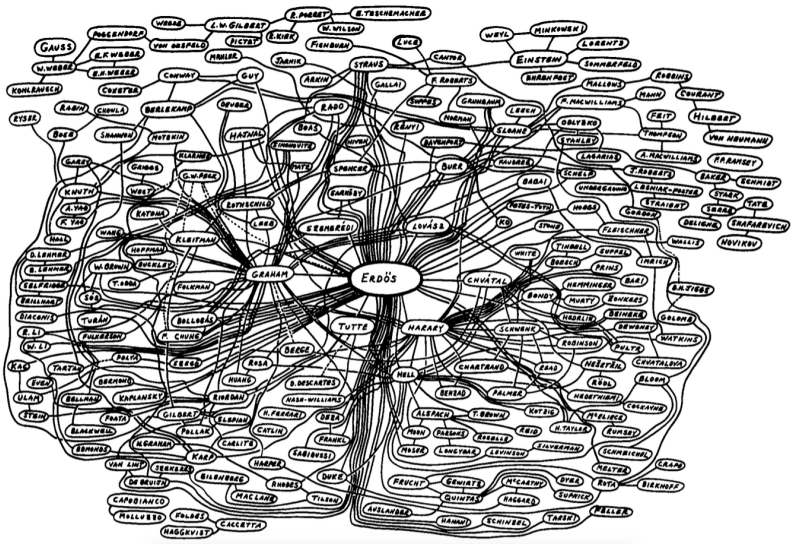



Figure 1.1: Ron Graham's hand-drawn picture of the collaboration network of the mathematician Paul Erdős. Nodes of the network account for mathematicians and a link between them is present if they co-authored a paper together. *Source: [164]. Reprinted with permission.*

biochemical properties of a cell. In Fig. 1.3 we show the network of metabolic interactions of the *Escherichia Coli* bacteria. Here nodes represent metabolites and there is a link between a pair of them if there is a chemical reaction in which one metabolite is a subtract and the other a product, or vice versa.

The three examples introduced give us an idea of what the appearance of networked data may be, as well as its potential to represent systems from many different fields. Besides representing this data in a graphic manner and obtaining a picture that is appealing to the eye, these systems can be exploited to extract a lot of information. A scientist, in possession of the toolset that has been curated for years and that now conforms network science, could ask a variety of different questions about those systems and obtain quantitative answers. Indeed, the analysis of the collaborative network of Paul Erdős and other collaboration networks, lead to pose the following questions: "what is the distance between two people in the network?"[88], "which is the average degree of separation between any pair of

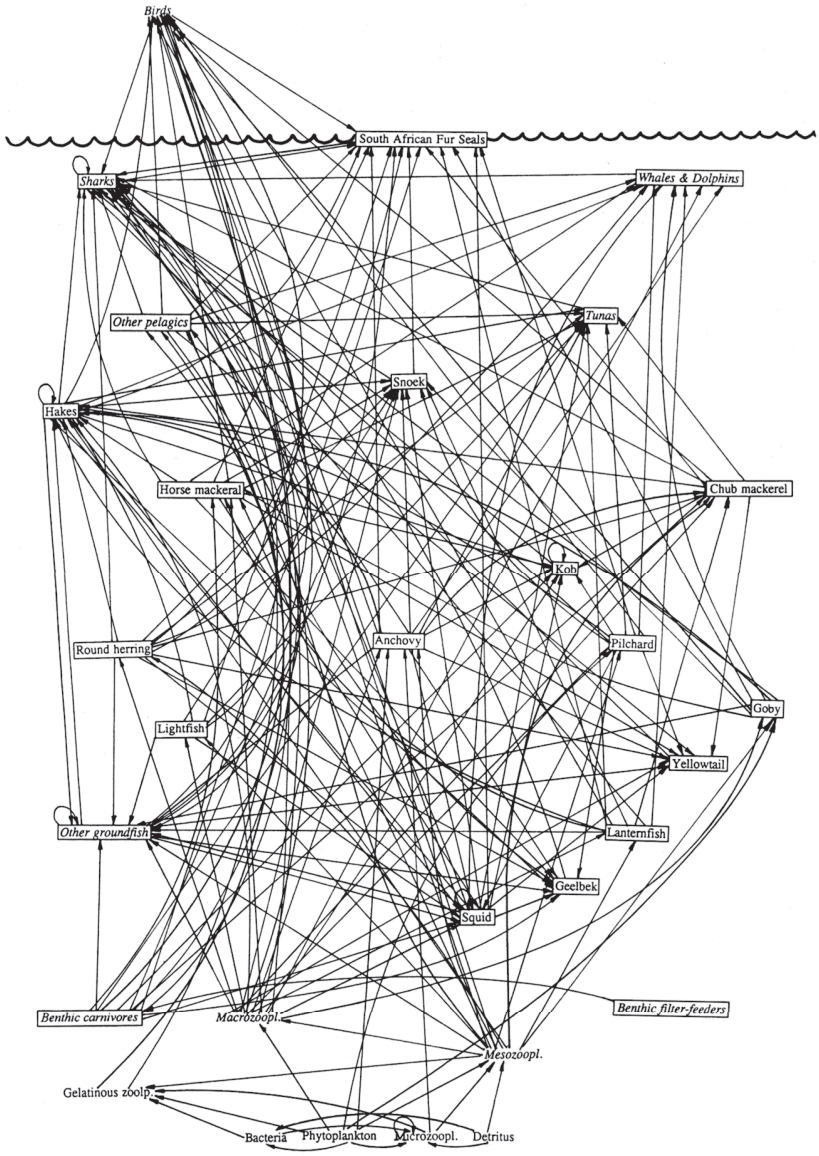



Figure 1.2: Subset of the food web of the Benguela ecosystem in South Africa. Nodes are animal species and a directed link goes from a prey to each of his predators. *Source: [219]. Reprinted with permission.*

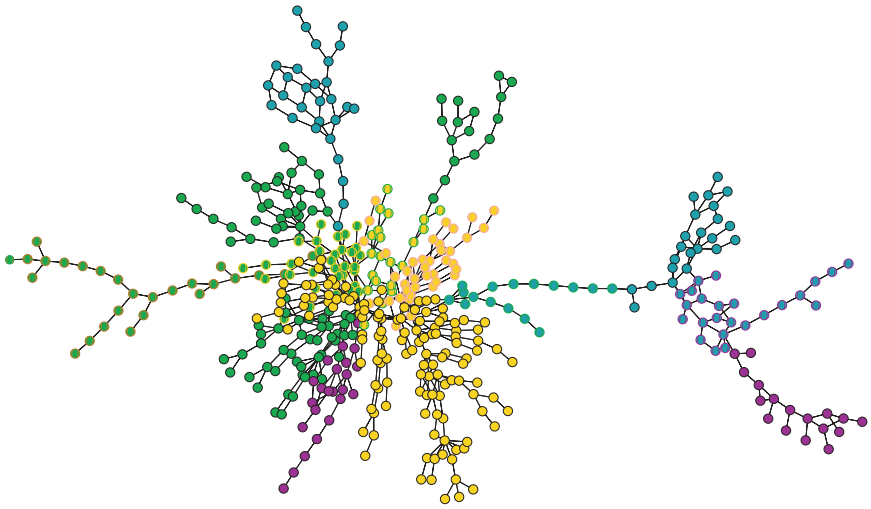


Figure 1.3: Metabolic network of E. coli, which contains 473 metabolites and 574 links. Each node is colored according to the module it belongs to. *Source: [99]. Reprinted with permission.*

scientists?" or "to what extent do collaboration networks show clustering?[1]"[153]. More generally, we could even ask "are social networks topologically different than other kinds of networks?"[161]. In the food web, we could ask "the removal of this particular species, to which extent will affect the other species?" or "which is the species whose extinction would put the whole ecosystem in danger?". In particular, in [219], the author studied the effect that killing seals may have on the species of fish that are caught for human consumption. Surprisingly, he found that even though seals are predators of this kind of fish, killing them would result in a even lower amount of fish in the markets. Also, a lot of interesting questions can be posed around metabolic networks. In particular, the authors of Fig. 1.3 aimed to find a subdivision of the network into groups, defined only by the topological features of the nodes of the network. Indeed, they divided the network into groups (represented by different colors in the figure) and furthermore, they found that nodes (metabolites) within the same group contributed to the same metabolic function [99].

1 The clustering coefficient is a measure of the degree to which nodes in a graph tend to cluster together. In practice, calculating the clustering coefficient of a network (global) implies counting the number of closed triplets over the total number of triplets [138].

In other words, if the scientist is able to gather networked data, then a never-ending amount of questions can be posed. What is the role of the individual nodes of the network? How can we characterize the structure? How does this structure influence the dynamics of the complex system that this network is representing? How does this pattern of interactions that I observe relate to the interactions I observe in other complex systems? The number of open problems arising from the observation of a single complex system is very large, and its extensive review exceeds the purpose of this introduction. In the following sections, however, I will explain some concepts about networks that must be introduced before digging deeper into the contents of this document.

## 1.2 MESOSCOPIC DESCRIPTION OF NETWORKS

The first works that paved the way into consolidating the field of complex networks were aimed to characterize the structure of the real networks we observe in nature. One of such works is the paper by the Faloutsos brothers [70], which analyzed the structure of the Internet and found that the degree distribution of the nodes in the network held a power-law. Numerous other works followed that one in the search of power-law distributed data in the biggest variety of topics, such as the frequency of occurrence of unique words in the novel Moby Dick [152], the number of citations received by scientific papers listed in the Science Citation Index in 1981 [179], or the number of species per genus of mammals [194], among many others[2]. Aside from trying to discern if the degree sequence of a network follows a power-law, the analysis of the large-scale structure of the network might also involve investigating the clustering coefficient, the path lengths distribution, the sizes of the components of the network or its connectedness. All these features are key for (I) having quantitative measures that then enable us to compare networks, (II) gain some insight on the dynamical processes that generated such networks, and perhaps more importantly, (III) designing generative models of synthetic networks that resemble those that we observe in the real world. Such models allow the scientist to understand the underlying mechanisms of network formation and growth, as well as being useful for simulating complex systems to which we do not have complete access. Up to date, the most widely used models are still the Erdős-Rényi model for random networks (which generates networks with degree following a Binomial distribution that strongly peaks at the average degree $\langle k \rangle$)[69], the Watts-Strogatz model for generating *small-world* networks (having high clustering and short path lengths)[218], and the Barabási-Albert preferential attachment model which generates networks exhibiting a power-law degree distribution [13].

Besides analyzing networks as a whole picture, another important perspective is the analysis at the node level. When a network is no longer considered a purely mathematical object but instead, its nodes and edges have a meaning, then getting insight into the individual properties of nodes of the network is interesting. For instance, if we study a social network of people and acquaintances, we might be interested in knowing which is the most important person in that network. Even though this question is understandable by everyone, the term

2 Later, the power-law character of some of these works was called into question in [41].

"important" lacks concreteness in a more formal context. Translating this into the mathematical domain, our question might be answered by looking for the node with highest degree (number of connections), which will surely represent an important actor in the network. The degree of the nodes is a first approximation to a measure of centrality, but there are others, such as the eigenvector centrality or the betweenness centrality [80]. The first measures how central is a node according to the centrality of his neighbors, and the second measures the extent to which a vertex lies on paths between other nodes. These measures can give us crucial information of our system, as for instance, in a network of routers and connections, the node with highest betweenness centrality will probably be the first to collapse if traffic flow increases [106].

As a matter of fact, we can calculate many descriptors of networks, either from the large-scale (macroscopic) perspective, or from the individual (microscopic) approach. Any of the descriptors introduced before would be of great use when trying to understand a little bit more about a complex system. However, networks can be analyzed at more than these two granularities. A very interesting point of view is what is called the *mesoscopic scale*, an intermediate length scale between the previous two, which accounts for analyzing the subgroups of the network. We are interested in these subgroups because intuitively, they are key to understanding the dynamics of the systems they describe. For instance, the most complex system we might ever know, the brain, is formed in its finest granular level, by neurons. In a very rough approximation, we may say that neurons tend to group together in bundles, and they are distributed along the volume of the brain in regions that are known to perform different functions. The brain as a whole exhibits a behavior so complex that it is impossible to approach all at once. Instead, by studying it at an intermediate scale, scientists were able to identify the main cognitive functions and their location. In short, studying systems at the mesoscopic level is a promising way to unravel the interplay between topology and dynamics. In the case of networks, this approach implies studying the groups of nodes and their potential role in the system's dynamics. Interestingly, these groups of nodes are not formed in an arbitrary manner, instead, they share topological properties. The problem of identifying these groups of nodes is a challenge by itself, usually referred to as *community detection*.

### 1.2.1 COMMUNITY STRUCTURE OF NETWORKS

In a nutshell, the aim of community detection is to find the groups of nodes in which the real networks we observe naturally divide, if there are any. We seek to find these groups because our intuition suggests that they might have an important role in the dynamics of such network. The first thing we have to consider then is deciding what exactly are we looking for; or, in other words, deciding what is a community. If we take a look at Fig. 1.4, which is a protein-protein interaction network of the rat proteome, we can see that it is divided into groups, which are also colored accordingly. At first glance, we can see that the communities of this network are formed by sets of nodes densely connected between them and not so connected with the other nodes in the network. This intuitive idea also works for many other kinds of networks, so it is acceptable to take it as our first definition of community. If we only had to detect the communities of networks as the one depicted here, it would be easy, simply taking a look at the picture and looking for groups that satisfy our condition would suffice. The problem is that finding groups of networks of hundreds or thousands of nodes requires using some kind of algorithm, which in turn, requires a more formal definition of community. So far, we have decided that communities are groups of nodes more tightly connected between them than with the rest of the network, but how much more tightly? Assuming that we find a mathematical condition that is able to describe our intuitive idea of community, then the problem of detecting communities loses sight of the actual meaning of the network and turns into a search problem. To solve it, we would design a (hopefully efficient) algorithm which would look for partitions of the network into groups of nodes that satisfy the mathematical condition we have imposed, and we would obtain one of these partitions as a result. If we discarded the actual meaning of the network, the problem would actually end at this point, as any subdivision of the network would be satisfying. However, as networks represent some sort of system, we might be curious to see if the groups we have found actually have any meaning at all. If the groups were known beforehand, then it would be easy to assess how good the result we obtained was. However, in the case where we didn't have any previous information about the system, then we would be obliged to rely solely on our algorithm and our own intuition. The latter situation is rather undesirable, and it stresses out the importance of evaluating the quality of the community structure found. Such evaluation can be done by designing measures to such purpose, or by making use of benchmarks. In our case, benchmarks are networks, either

synthetic or real, whose division into groups is known. These benchmarks are used to put the community detection algorithms to test, in order to provide the scientist some clue about their performance. Knowing beforehand the behavior and possible flaws of these algorithms is necessary to understand the outcome obtained by applying them. In the next sections we get into more detail on the problem of defining communities, algorithms to find them and quality functions.

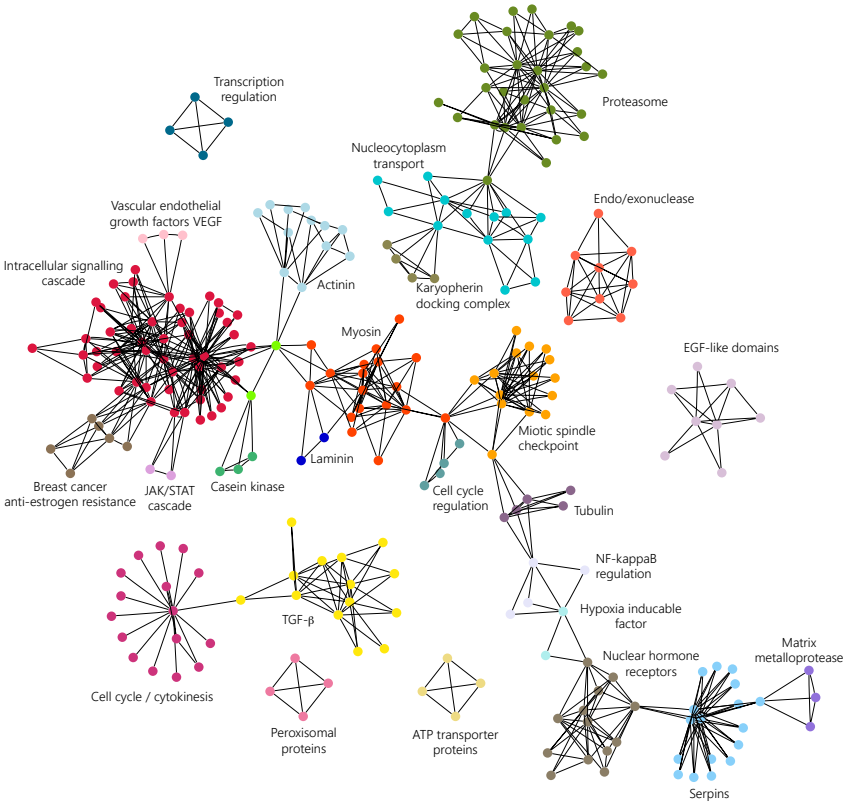



Figure 1.4: Representation of a protein-protein interaction network corresponding to the rat proteome. The links have been derived from many experimental observations in different organisms. The colors correspond to well-known functional groups according to biological knowledge of the system. *Source: [115]. Reprinted with permission.*

#### 1.2.1.1 *Definitions of community*

The problem of community structure, intuitive at first sight, is actually ill-posed. As we may notice, the concept of community is not rigorously defined, and furthermore, its definition often depends on the specific system that we are analyzing. In the literature there are plenty of different approaches and recipes, an extensive review can be found in [76]. In general, definitions of communities can be classified into two groups: local and global definitions. On one side, *local definitions* take into account nodes and their neighborhood, while global definitions use information of the whole structure of the network. The first approach to defining what is a community was to consider the parallelism between communities and cliques, proposed in [138]. Obviously, cliques inside a network should correspond to communities, but the requirement in the other direction is too restrictive. A clique that lacks a link is also a very cohesive structure, but in this case it wouldn't satisfy our condition. A second proposal of definition, used in [3, 137], was to define a community as the maximal subgraph such that the distance between each pair of its vertices is not larger than a certain quantity. However, these two definitions are missing a key point of the concept of communities: even though the subgraphs have to be internally cohesive, they wouldn't be communities if there was the same cohesion to the rest of the network. To account for this issue, more recent definitions of community were based on comparing the internal and the external connectivity of groups. A first definition of strong community, which is used in [74], is that all nodes in the subgroup $C$ should fulfill the following criterion:

$$k_i^{\text{in}}(C) > k_i^{\text{out}}(C), \forall i \in C, \tag{1.1}$$

where $k_i^{\text{in}}$ and $k_i^{\text{out}}$ are the internal and external degrees of nodes $i$. This strict definition requires each single node in the community to have higher internal degree than external. On the other hand, the weak definition only requires that the *sum* of the degrees of nodes in the community has to be higher internally than externally:

$$\sum_{i \in C} k_i^{\text{in}}(C) > \sum_{i \in C} k_i^{\text{out}}(C), \forall i \in C. \tag{1.2}$$

On the other side, *global definitions* of communities define them with respect to the graph as a whole. A possibility would be, for instance, comparing the graph at study with a random graph with equivalent topological descriptors. A random graph is expected to have low variations in the mean degree and therefore, no

community structure whatsoever. By comparing the random null-model to our network we can discern if our network shows community structure. This is the idea behind modularity, as we will see in a moment.

#### 1.2.1.2 *A hard problem*

Assuming that we have found a satisfying mathematical definition of community, the second problem is to design an algorithm that looks for such structures. Unluckily, this step is not so straightforward. Consider the simplest exercise possible, in which we would hypothetically be interested in dividing a network in two pieces of given sizes such that the number of edges between the groups (also known as *cut size*) is minimum. One could try to look for all possible groups of nodes of the desired sizes and keep the configuration that minimizes the cut size. This is doable in the case of very small networks or very dense ones, but in realistic scenarios this exhaustive approach is prohibitive in terms of computation time. Indeed, the number of ways in which we can divide a network in two groups of $n_1$ and $n_2$ nodes is $n!/(n_1!n_2!)$, which, approximated using Stirling's formula, we get:

$$\frac{n!}{n_1!n_2!} \simeq \frac{n^{n+1/2}}{n_1^{n_1+1/2}n_2^{n_2+1/2}}. \tag{1.3}$$

If, for instance, we wished to divide our network into two equally sized groups of $\frac{1}{2}n$ each, the number of ways of doing it is:

$$\frac{n^{n+1/2}}{(n/2)^{n+1}} = \frac{2^{n+1}}{\sqrt{n}}, \tag{1.4}$$

which implies that the amount of time needed to go through all configurations grows exponentially with the size of the network $n$.

In community detection, we do not ask for a pre-specified number of groups, neither do we put restrictions on their sizes, which makes the problem even tougher. In fact, the number of ways to divide $n$ vertices into $g$ non-empty groups is given by the Stirling number of the second kind $S_n^{(g)}$, and hence the number of distinct community divisions is $\sum_{g=1}^{n} S_n^{(g)}$. This sum is not known in closed form, but we observe that $S_n^{(1)} + S_n^{(2)} = 2^{n-1}, \forall n > 1$, so the sum must increase at least exponentially with $n$.

In practice, to get around the fact that we are not able to perform an exhaustive search in the space of possible configurations, we use heuristics. Heuristics are good approximations to the solution that we aim to find, cleverly designed to avoid having to look for too many configurations. Of course heuristics cannot ensure that your result is the best possible, but normally they do an acceptable job.

#### 1.2.1.3 *Traditional approaches to community detection*

Two very tightly related approaches to the problem of dividing a graph into groups are *graph partitioning* and *community detection*. Graph partitioning has been studied in computer science and related fields, having in mind applications in parallel computing and integrated circuit design. Community structure, as we call it –or block modeling or clustering, as referred to by other fields– has been studied mainly by sociologists, biologists, applied mathematicians and physicists, with applications in biological and social networks. We could say they address the same question, albeit by somewhat different means. Graph partitioning aims to divide a network into an specific number of groups of fixed size. A typical example of graph partitioning is the division of computational tasks between the different processors in a parallel computer, with the purpose of balancing the load and minimizing the communication between processors. In this example, the parameters are fixed because we know beforehand the number of processors we have and its load capacity. Community detection however, has a different purpose: dividing the network into the naturally present groups to gain understanding of that system. No parameters are fixed and if the network were not to display any community structure at all, the method would precisely return that, and this would be of enough interest for the light it sheds on the topology of the network. In this document, we focus on community detection. However, the classical approaches to graph partitioning are useful and worth mentioning, as they motivated some of the currently used community detection algorithms.

In graph partitioning, we are interested in dividing a graph into a particular number of groups of a desired size. Normally, the separation into more than two groups is achieved by repeated *graph bisection*, which means dividing the graph in two. If we wished to split it in three equally sized groups, we would look initially for a partition in two groups, one having double size than the other. After that, we would split the big group in two, achieving the three desired groups. The most famous algorithm to approach the problem of graph bisectioning is the *Kernighan-*

*Lin algorithm* [121]. Roughly stated, this algorithm divides the network into two groups of the required sizes, and initially distributes the nodes randomly in the two groups. Then, taking a node $i$ from one group and a node $j$ from the other, it calculates how much the cut size would diminish if those two nodes were exchanged. It does this for all pairs of nodes in the network, and chooses the pair that minimizes the cut size, with the restriction that already swapped pairs cannot be swapped again. This approach is fairly effective, although quite slow, due to the number of swaps that have to be done. A faster algebraic method for partitioning networks is the *spectral partitioning* method of Fiedler [72, 176], which makes use of the matrix properties of the graph Laplacian.

A classical method that serves the purpose of detecting communities is known as *hierarchical clustering*. Here, we first calculate a weight $w_{ij}$ for each pair of vertices in the network, which accounts for how closely connected they are. The weights can be calculated in very different ways, for example, counting the number of node-independent[3] paths between the two nodes. Then, one starts out from the isolated nodes in the network, and starts adding the edges between them, in decreasing order of their weights. As we add edges, the graph shows a nested set of increasingly large components, which we take as our communities. To represent all the nested configurations, we may plot a dendrogram, where each horizontal slice is a partition. This method is useful, albeit far from perfect. Firstly, because it tends to separate peripheral vertices in single communities instead of attaching them to the community they are closer to. Second, because although having the whole nested structure is informative, there's no assessment on the partition we might take as our result.

#### 1.2.1.4 *Girvan & Newman approach to community detection*

In 2002, Girvan & Newman proposed a divisive algorithm for detecting communities [85]. It is in some way related to the approach of hierarchical clustering, but in this case they take a "top down" approach: all observations start in one cluster, and splits are performed recursively as one moves down the hierarchy. Additionally, instead of constructing a measure to evaluate which edges are more central to communities, they look for the edges that are least central, the ones that bridge communities. They generalize the measure of vertex betweenness centrality of Freeman [81], to a measure of *edge betweenness centrality*. According

3 Two paths that connect the same pair of vertices are said to be node-independent if they share none of the same vertices other than their initial and final vertices.

to their definition, the betweenness of an edge is the number of shortest paths that run through it. The intuition behind the use of low betweenness edges in detecting communities is clear: if communities are densely connected inside but loosely connected between them, then the nodes that join communities will have high betweenness. The process then consists in calculating the betweenness for all edges, and removing the edge with the highest score. After, betweenness has to be recalculated for all edges affected by the removal, and the process repeats itself until no edges remain. This process results, as in the case of hierarchical clustering, in a hierarchy of nested communities. Again, choosing a partition from all the obtained ones means choosing an horizontal cut in the resulting dendrogram, without any preference for choosing one over the other. They tested their method against synthetic networks designed for such purpose, as well as with real networks. The results were encouraging, see the results for the Zachary Karate Club network [221], in Fig. 1.5 (a).

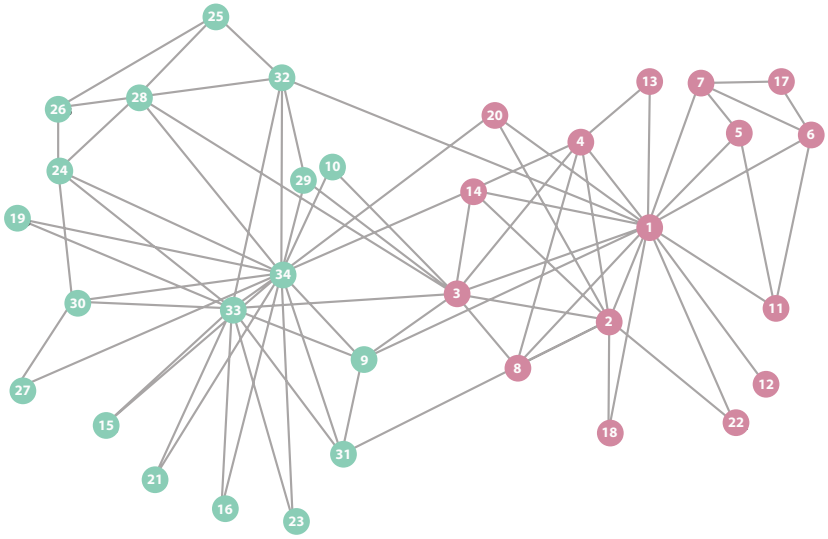
This network has become a classic example for community detection, and consists of 34 nodes that account for the members of a karate club, which Zachary studied for a period of two years. At one point a disagreement between the club's owner and the instructor led to the instructor leaving and starting a new club, and some of the old members followed him. Discovering the two groups from the networked data is the aim of any community algorithm that uses this network as a test. As we can see in Fig. 1.5 (b), the edge betweenness method is able to almost perfectly detect this structure, as the first split in the dendrogram divides the network in the two desired groups, except for the misplacement of node 3.

As encouraging as the results may be, there's still a missing piece. Having had no information about the real groups in the Zachary network, which one of the levels of the dendrogram would we have taken as solution? We need a form to evaluate the partitions obtained by this or any algorithm, to have an assessment on the *quality* of such partitions.

### 1.2.2 MODULARITY

In most practical situations we will not know the structure of communities beforehand. Moreover, algorithms such as hierarchical clustering or the edge betweenness method presented before, deliver a good amount of partitions. The problem that is raised is then: how can we identify "good" communities? The measure introduced by Newman and Girvan in [160] known as *modularity* aims

a)




b)

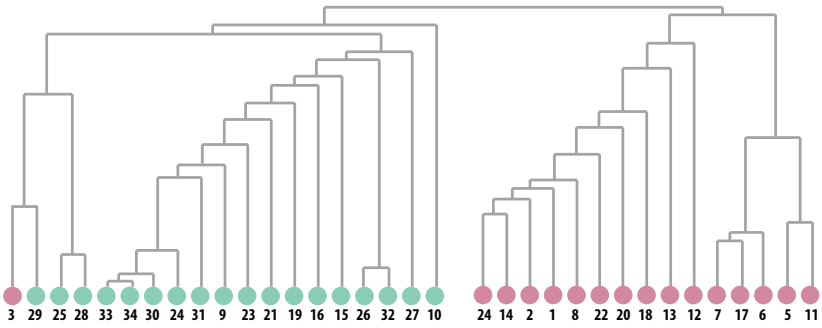



Figure 1.5: **(a)** Representation of the unweighted version of the Zachary Karate Club network. Differently shaped labels of nodes account for the *real* division of the network in two communities. **(b)** Dendrogram representing the hierarchy of communities obtained with the Girvan & Newman method of edge betweenness. *Source: High resolution picture created by the author of this document resembling the one in [85].*

at answering that question. The calculation of modularity for a given partition of an unweighted, undirected network is as follows. Consider a division of the network in $c$ communities. Let us define a matrix $\mathbf{e}$ of size $c \times c$ whose element $e_{rs}$ is the fraction of the edges in the network that link vertices in community $r$ to vertices in community $s$. The trace of the matrix $\mathrm{Tr}\, e = \sum_r e_{rr}$ gives the fraction of intra-community links[4], which should be high with respect to $e_{rs}$ for networks with clear community structure. Additionally, we need to define the row (column) sums $a_r = \sum_s e_{rs}$, which account for the fraction of links attached to nodes in community $r$. If the edges of the network were connected randomly, then we would have $e_{rs} = a_r a_s$. Therefore, modularity is defined as:

$$Q = \sum_r (e_{rr} - a_r^2) = \mathrm{Tr}\, \mathbf{e} - \|\mathbf{e}^2\|. \tag{1.5}$$

This quantity measures the amount of intra-community links minus the expected number of those links that we would have if the network were randomly rewired. Note that the larger the value of $Q$, the more community structure it has compared to the random case. The values of $Q$ are bounded by definition between $[-\|\mathbf{e}^2\|, 1)$. Using this measure to assess the partitions we should keep in the case of hierarchical clustering or any algorithm producing a tree of results, implies calculating the value of $Q$ for each level of the resulting dendrogram. The "best" partition corresponds to the maximum of these values.

An alternative (but equivalent) expression for modularity, which is the one used in [156] and in the rest of this document is:

$$Q(C) = \frac{1}{2m} \sum_i \sum_j \left( A_{ij} - \frac{k_i k_j}{2m} \right) \delta(C_i, C_j)\,, \tag{1.6}$$

where $m$ is the number of links in the network, $A$ is the adjacency matrix whose entries $A_{ij}$ are 1 if there's a link between $i$ and $j$ and zero otherwise, $k_i$ is the degree of node $i$, $C_i$ refers to the community to which node $i$ belongs, and the Kronecker delta function $\delta(C_i, C_j)$ equals 1 if nodes $i$ and $j$ are in the same community and zero otherwise. The generalization to weighted networks [156] is:

$$Q(C) = \frac{1}{2w} \sum_i \sum_j \left( W_{ij} - \frac{w_i w_j}{2w} \right) \delta(C_i, C_j)\,, \tag{1.7}$$

4 Links that connect nodes inside the same community.

where now $w$ is the sum the weights of the weighted adjacency matrix $W_{ij}$ and $w_i$ is the strength of node $i$. Obviously, the semantics remains the same.

#### 1.2.2.1 *Modularity optimization*

The purpose of modularity is to evaluate the quality of the partitions obtained by community detection algorithms, where high values of modularity indicate good partitions, meaning those that we would be unlikely to find if the network was connected at random. This opens up an alternative approach to finding community structure: why don't we get rid of the community detection algorithm and instead, directly optimize modularity Q over all possible divisions to find the best one? This was the proposal made by Newman in [157], where he also introduced an algorithm for such purpose. The first difficulty encountered is that the cost of calculating $Q$ for all possible partitions in the network is prohibitive. As we mentioned, an exhaustive search of all possible configurations in groups would take an amount of time at least exponential in the number of vertices, this is why we need to use heuristics to such purpose.

The proposal by Newman in the aforementioned paper, known as the *fast algorithm*, is an agglomerative hierarchical clustering method. The initial setup is a partition of the network where each single node is assigned to a different community, thus there are $n$ communities to start with. Then, communities are joined repeatedly in pairs, choosing each time the pair of communities whose union results in the greatest increase (or smallest decrease) in the quantity $Q$. Even though this process also provides with a hierarchy of partitions, the problem of choosing one of them is easily addressed by keeping the one with highest modularity. This algorithm has the advantage of being relatively fast, since the time necessary to complete the dendrogram is $O(n^2)$

Other available methods for optimizing modularity are briefly commented here:

- *Greedy techniques.* They perform a local search in the space of solutions taking the best available solution at each step. The fast method of Newman falls in this category. Other algorithms that use greedy procedures to optimize modularity are [23, 188, 85].
- *Simulated annealing.* A stochastic greedy approach in which the search on the space of solutions is not always optimal at each step. Such methods are controlled by a *temperature* parameter, which determines the probability of accepting a solution of the space of partitions that does not improve the last

solution found. The temperature goes to zero as the algorithm evolves. The advantage in front of greedy algorithms relies on the relative independence of the initial state on the final outcome. Simulated annealing can be used for average sized graphs, with up to about $10^4$ vertices. A work in which modularity is optimized using this technique is [99].

- *Extremal optimization.* As an alternative for simulated annealing, Boettcher and Percus proposed an heuristic with an accuracy comparable to simulated annealing but which takes substantially less computational time [26]. This technique was first used to optimize modularity in [64], where its basis is to calculate locally the contribution of each vertex to the global modularity. Starting out from a random partition of the graph in two equally sized groups, vertices with lower contribution to modularity are shifted to the other community. This is done repeatedly, along with recalculating $Q$ at each step, until no further improvement of modularity can be achieved. This method is used in some of the works that I will present in this document.
- *Spectral optimization.* Modularity can be optimized by calculating the eigenvectors and eigenvalues of the modularity matrix $B$, defined as $B_{ij} = A_{ij} - \frac{k_i k_j}{2m}$. One has to look for the eigenvector of $B$ with the highest positive eigenvalue, and group the vertices according to the signs of its components. Spectral optimization of modularity was addressed in [158, 159].
- *Tabu optimization.* Based on the Tabu search [87], its adaptation to the optimization of modularity was presented in [10]. The main idea is to start out from a random partition of the network. Then, we calculate the "neighbor partitions", defined as the set of partitions that would result from moving one node from a community to another, in a random fashion. Calculating the modularity for each neighbor partition, we keep the one with highest score and discard the others, and the process starts over again. Additionally, we keep an updated list of the moves that we have already done (the *tabu* list), which will not be allowed in the subsequent steps of the procedure, in order to avoid getting trapped in local optima or cycles.

#### 1.2.2.2 *The resolution limit of modularity*

The success of modularity as a quality function to analyze the modular structure of complex networks, relies on its intrinsic simplicity. The researcher interested in this analysis is endowed with a non-parametric function to be optimized: mod-

ularity. The result will be a division of the network into communities which the scientist will consider as a good one if the value of $Q$ obtained is high. However, as we will see, it has been shown that modularity is not the panacea of the community detection problem; in particular it suffers from a resolution limit that avoids grasping the modular structure of networks at some scales of resolution.

It is worth reminding that when the problem of detecting communities is translated into the mathematical domain, it implies accepting this new more formal definition of community, and losing sight of our intuitive idea of what communities look like. When a scientist chooses an algorithm to detect communities, he must know what is the definition of community that comes with it, and re-adapt his expectations on what the outcome of such algorithm would be. In the case of an algorithm that optimizes modularity, we should embrace that the definition of community is no longer "tightly packed groups of nodes", but instead, "dense substructures that were not likely to happen by random chance". By no means this implies that we should blindly accept any result provided by any available algorithm, as each of them is designed differently and of course some design decisions can lead to poor results. As scientists, our work is to be critics with any finding that contradicts our scientific intuition, but also we have to know when we are fighting against the elements. In the particular case of modularity, some work has been done in discovering situations where the results obtained are correct according to the definition, but truly counter-intuitive.

The first problem is that of modularity finding partitions in random networks. In random graphs, vertices are linked with a probability either constant or as a function of their degrees, so in principle there shouldn't be groups of nodes with a special connectivity. The problem, as pointed out in [181, 177], is that there may be fluctuations in the distribution of edges in the graph, which result in a non-homogeneous degree distribution. These fluctuations provoke high concentrations of links in some parts of the graph, which modularity interprets as communities.

The second problem, pointed out in [78], is that of modularity dividing a graph with an intuitively very clear community structure in groups different than expected. Imagine two groups of nodes, $\mathcal{A}$ and $\mathcal{B}$, with degrees $k_{\mathcal{A}}$ and $k_{\mathcal{B}}$ respectively. The number of expected links between these two groups is then $p_{\mathcal{A}\mathcal{B}} = k_{\mathcal{A}}k_{\mathcal{B}}/2m$. The variation of modularity when merging $\mathcal{A}$ and $\mathcal{B}$ in a single community with respect to the partition where each one forms its own community is $\Delta Q_{\mathcal{A}\mathcal{B}} = l_{\mathcal{A}\mathcal{B}}/m - k_{\mathcal{A}}k_{\mathcal{B}}/2m^2$, where $l_{\mathcal{A}\mathcal{B}}$ is the number of edges between $\mathcal{A}$ and $\mathcal{B}$. For the sake of simplicity, let's suppose that $k_{\mathcal{A}} \sim k_{\mathcal{B}} = k$ (both groups

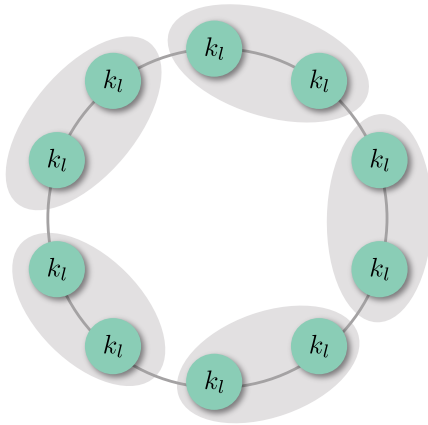



Figure 1.6: Representation of a toy network that suffers from the resolution limit of modularity. Here, each node is a clique of $l$ vertices. In a setup where the number of cliques is larger than the square of its size, modularity delivers communities containing pairs of cliques. *Source: High resolution picture created by the author of this document resembling the one in [78].*

have then the same number of edges), and that there is only one link between the two groups, $l_{\mathcal{A}\mathcal{B}} = 1$. If $k < \sqrt{2m}$, then the contribution to modularity is greater if they are joined in the same cluster. The intuition is clear: modularity will merge groups that are connected with an amount of links higher than expected. If the subgraphs are sufficiently small (in degree), then the expected number of edges between the two groups in the null model can be smaller than one, which translates into a single link being able to keep the two graphs together. In general, this feature is not undesirable, but this result holds independently of the structure of subgraphs, which means it is also true in the case where the two groups are cliques, the most cohesive structures possible. To illustrate this particular scenario, I make use of the example provided in [78], which consists in a network of $n_c$ identical cliques, with $l$ vertices each, connected by single edges. Again, our intuition would say that the communities should correspond to the single cliques. However, if $n_c$ is larger than $l^2$, the modularity contribution would be higher if the cliques were joined in pairs, as depicted in Fig. 1.6.

What we learn from this is that modularity presents a resolution limit, beyond which we are not able to discern the community structure of the network. It will not properly detect tightly connected clusters of a size comparatively small with

respect to the whole network. In general, quality functions based on null-models such that the horizon of the vertices is of the order of the size of the network are likely to be affected by a resolution limit of this kind [75]. More recent findings [148, 57, 145, 139, 58], suggest the presence of a phase transition in matrix methods for detecting communities. This transition separates a regime in which such methods successfully detect the community structure from one in which the structure is present but is not detected. This suggests that the resolution limit is not a particular feature of modularity, instead, is a common flaw shared by all community detection methods based on global quality functions.

### 1.2.3 MULTI-RESOLUTION MODULARITY OPTIMIZATION ALGORITHMS

Due to the recent findings about the resolution limit of modularity, some scientists have tried to adapt the classical methods for community detection to be able to diminish the effect of this limit of resolution. To do so, one of the most well-known techniques is to try to analyze the community structure at different levels of resolution.

#### 1.2.3.1 *Motivation*

Facing a famous painting by Salvador Dali (see Fig. 1.7) we can observe that in this painting, as it happens in complex networks, there is not only one scale of resolution which may be interesting to analyze, but there are many which coexist at the same time and contain relevant information. If we observe the painting close enough, we can see that the picture is actually formed by small tiles which have drawings in them. They are the minimum unit of the painting and altogether they form the microscopic structure. Instead, if we place ourselves 20 meters from the painting, as the author suggests, what we see is that all those tiles join to form the face of Abraham Lincoln. In that position, we would observe the macroscopic structure of the system. However, there exist intermediate scales of resolution between the microscale and the macroscale, which conform the mesoscale. In one of those intermediate scales we can see what the author is trying to represent, which is his wife Gala, looking at the sea through a window.

Relating this to the problem of community detection, we can say that multi-resolution algorithms are those that deliver, not only a single partition of the network, but the whole mesoscopic structure. As a bonus, such algorithms also

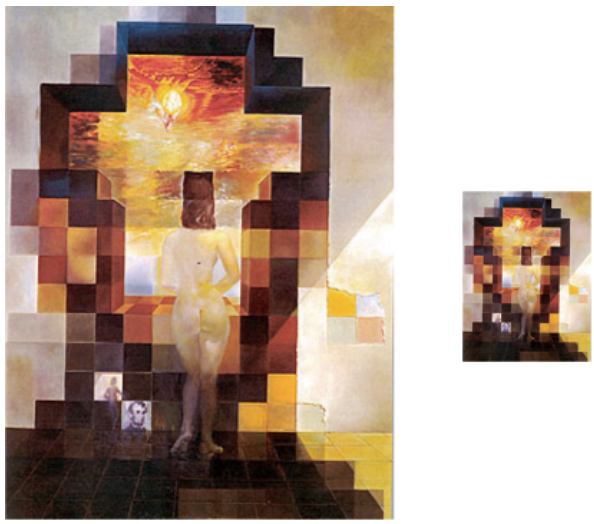


Figure 1.7: *"Gala contemplating the Mediterranean sea which at twenty meters becomes a portrait of Abraham Lincoln"*, by Salvador Dali, 1974. Left, at shorter distance, and right, at longer distance.

serve the purpose of palliating the resolution limit of modularity, as having access to more than one scale of resolution allows us to grasp even those particular communities that, as we have shown, are sometimes apparently unaccessible.

#### 1.2.3.2 *The Arenas-Fernández-Gómez algorithm*

The Arenas-Fernández-Gómez (AFG) algorithm, is the mainstay of some of the contributions made by the author of this document and presented in the next chapter. Introduced in [10], this algorithm adds a resolution parameter to the formulation of modularity which, when tuned, is able to give us access to the whole mesoscale of the network. The resolution parameter introduced is a self-loop for each node, with equal value $r$ for all nodes. The self-loop only affects the diagonal of the (weighted) adjacency matrix, and therefore the main statistical properties of the network (degree distribution, degree-degree correlation, spectrum, etc) are preserved.

The formulation of the AFG algorithm for the case of weighted networks is explained below. First of all, let us rewrite Eq. 1.7 in terms of the contributions of the modules instead of nodes:

$$Q = \sum_{s=1}^{m} \left( \frac{w_{ss}}{w} - \left( \frac{w_s}{2w} \right)^2 \right), \tag{1.8}$$

where we sum over the $m$ modules of the partition, $w_{ss}$ is the internal strength of module $s$ and $w_s$ the total strength of module $s$. For unweighted networks $w_{ss}$ reduces to the number of internal links and $w_s$ to the sum of degrees of the nodes in module $s$, thus leading to Eq. 1.5.

The problem now is how to increase the strength of nodes without altering the topological characteristics of the original network. The problem is solved by rescaling the topology: starting out from the original weighted adjacency matrix $\mathbf{W}$, they define $\mathbf{W}_r$ as:

$$\mathbf{W}_r = \mathbf{W} + r\mathbf{I}\,, \tag{1.9}$$

where $\mathbf{I}$ is the identity matrix. In terms of graphs, this new matrix represents the original network with self-loops of weight $r$ for every node. Note that the prescription in Eq. 1.9 supposes a constant shift (translation) $r$ of the strength of each node.

Denoting $Q_r$ the modularity of the network at scale $r$, the equivalent expression to Eq. 1.8 reads

$$Q_r = \sum_{s=1}^{m} \left( \frac{2w_{ss} + n_s r}{2w + Nr} - \left( \frac{w_s + n_s r}{2w + Nr} \right)^2 \right). \tag{1.10}$$

We may also express the latter equation in terms of contributions of nodes, instead of modules, which is possibly the most commonly used notation:

$$Q_r = \frac{1}{2w + Nr} \sum_{i,j} \left[ (w_{ij} + r\delta_{ij}) - \frac{(w_i + r)(w_j + r)}{2w + Nr} \right] \delta(C_i, C_j) \tag{1.11}$$

The parameter $r$ accounts for *resistance parameter*, which should be interpreted as the resistance a node has to form part of a community. Loosely speaking, when the resistance parameter is positive and has its maximum value ($r_{\max}$), the nodes

are reinforced and each would form its own community. This means that the partition obtained is dividing the network in singletons. On the other case, if the value of $r$ is negative and has its minimum value ($r_{\min}$), the nodes are forced to attach to other nodes in the network to form a community, therefore the partition obtained is formed by a single community which contains all nodes. Tuning this parameter from $r_{\min}$ to $r_{\max}$ and performing a modularity optimization at each step, we are provided with a partition for each level of resolution of the network and we are able to observe the whole mesoscale as intended. Note that the original formulation of modularity is recovered for $r = 0$.

To illustrate the performance of this algorithm, the authors use the synthetic network in Fig. 1.8 (a). The benchmark is a network of 256 nodes, homogeneous in degree with two predefined hierarchical levels. Each node has 13 links within the smaller communities, 4 links within the big communities, and 1 link connecting this node to a node in any of the three other big communities. Two correct solutions of the problem of community detection coexist: the division in 4 or 16 communities. The application of the AFG algorithm on this benchmark leads us to the results shown in Fig. 1.8 (b). There, the authors plot the number of modules of the partitions obtained as a function of the resistance parameter $r$. As the algorithm delivers as many partitions as values of the resistance, a way to choose the partition we keep is to look which one of them remained unchanged for more values of $r$. We can see that there are two plateaus in the figure, which correspond to the division of the network in four communities (Fig. 1.8 (b)(I)) and the division in 16 communities (Fig. 1.8 (b)(II)). Other configurations are also detected but are not stable. It is worth emphasizing that using the traditional formulation for modularity, where $r = 0$, we would have had access only to the division in four groups.

#### 1.2.3.3 *The Reichardt-Bornholdt algorithm*

There are other multi-scale methods for optimizing modularity. Reichardt & Bornholdt (RB) [180] reformulated the problem of community detection as the problem of finding the ground state of a spin glass model. The authors assign each vertex $i$ with a Potts spin variable $\sigma_i$, which represents an indicator of the community to which the node $i$ belongs. The main idea is that nodes with the same spin state should probably have a link between them and therefore be placed in the same community, and vice versa. The advantage of this approach is that the problem of maximizing modularity is translated into the problem of

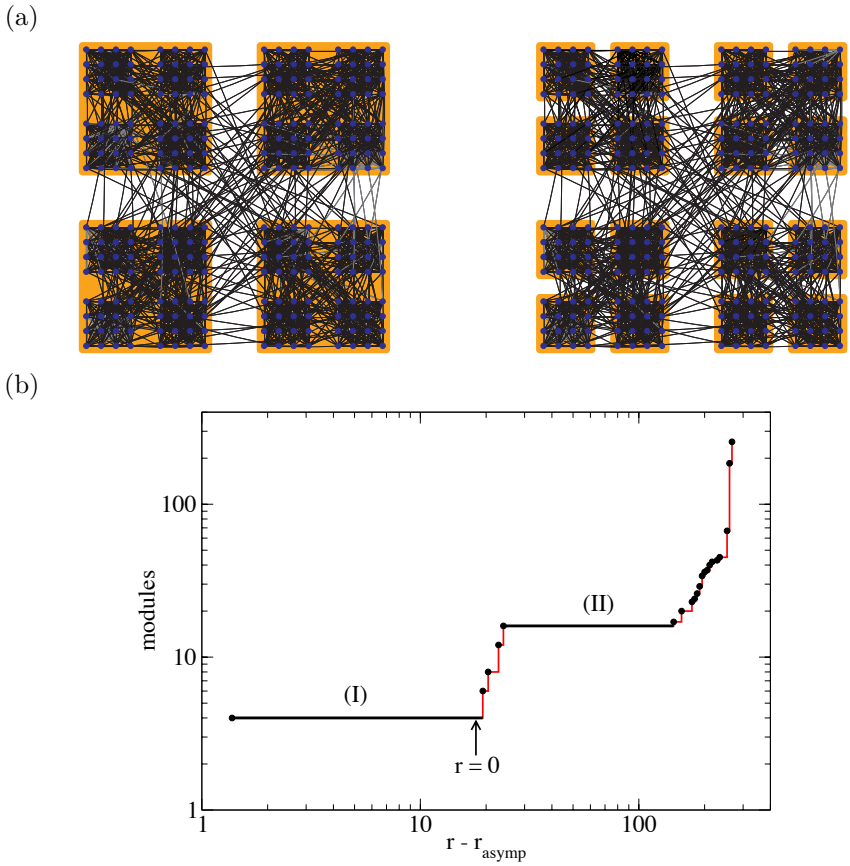



Figure 1.8: (a) Synthetic network that presents internal community structure at different scales, formed by 256 nodes. Left, partition of the network into four communities of 64 nodes each. Right, division into 16 communities of 16 nodes each. (b) Results of the application of the AFG algorithm. Plot of the number of communities found at each value of the $r$ parameter. We observe that the two scales that last the most correspond to the two expected divisions of the network *Source: [10]. Reprinted with permission.*

minimizing the energy of the Hamiltonian, subject which plenty of literature has addressed. The Hamiltonian of the model is:

$$\mathcal{H}(\{\sigma\}) = -\sum_{i<j} J_{ij}\delta(\sigma_i,\sigma_j) = -\sum_{i<j} J(A_{ij}-\gamma p_{ij})\delta(\sigma_i,\sigma_j), \tag{1.12}$$

where $J$ accounts for the coupling strength, $A_{ij}$ are the elements of the adjacency matrix, $\gamma > 0$ expresses the relative contribution to the energy from existing and missing edges, and $p_{ij}$ is the expected number of links connecting $i$ and $j$ for a null model graph with the same total number of edges $m$ of the graph at study. The parameter $\gamma$ introduced in their model adds the multi-resolution behavior to the model. By tuning this parameter, one can vary the number of clusters in the partition with minimum energy, ranging from a single cluster containing all vertices ($\gamma = 0$), to the partition where all nodes are in the same cluster ($\gamma \to \infty$). The formulation behind this concept, adapted to modularity optimization, reads:

$$Q_\gamma = \frac{1}{2w}\sum_i\sum_j \left(w_{ij} - \gamma\frac{w_i w_j}{2w}\right)\delta(C_i,C_j)\,, \tag{1.13}$$

The purpose of tuning the $\gamma$ parameter is to change the null term at each step, thus obtaining different resolutions for each $\gamma$ value. This method and the AFG algorithm presented previously are not equivalent, so there is not a translation between $\gamma$ and $r$. For a more detailed comparison between these two methods, see [10].

#### 1.2.3.4 *Other multi-resolution modularity optimization algorithms*

Other multi-resolution modularity optimization algorithms, that are not addressed in the contributions of this thesis, but are worth mentioning, are summarized next.

*Pons and Latapy* proposed a method in [175], which introduced another multi scale formulation[5] for modularity:

$$Q_\alpha^{\mathcal{M}} = \sum_{c=1}^{n_c}\left[\alpha\frac{l_c}{m} - (1-\alpha)\left(\frac{d_c}{2m}\right)^2\right]. \tag{1.14}$$

5 This notation is equivalent to the *Reichardt and Bornholdt* one, with the parameter $\gamma$ rescaled: $\gamma = \frac{1-\alpha}{\alpha}$.

Here, the resolution parameter is $0 \leq \alpha \leq 1$, where $\alpha = 0$ corresponds to smallest communities with only one vertex and $\alpha = 1$ corresponds to the largest community containing all the vertices. For $\alpha = 1/2$, standard modularity is recovered. They also propose a function to evaluate the relevance of the partitions, for any given multi-scale quality function. They suggested that the length of the $\alpha$-range $[\alpha_{\min}(\mathcal{C}), \alpha_{\max}(\mathcal{C})]$, for which a community $\mathcal{C}$ belongs to the maximum modularity partition, is a good indicator of the stability of the community.

*Ronhovde and Nussinov* proposed in [183] a technique based on the Potts model, similar to the RB approach. The Hamiltonian is:

$$\mathcal{H}(\{\sigma\}) = -\frac{1}{2}\sum_{i \neq j}[A_{ij} - \gamma(1 - A_{ij})]\delta(\sigma_i, \sigma_j) \tag{1.15}$$

The main difference with the RB approach is that here the null model term is constant. Mainly, for a division into communities, this formulation rewards the existence of edges between pairs of nodes in the same community and penalizes the non-existence. The $\gamma$ parameter decides the tradeoff between the two contributions. The algorithm works by swapping pairs of nodes to the communities where they lead to a largest decrease in the system's energy. Another interesting feature of this algorithm is that for each vertex, only its neighborhood is explored. As all the merging/splitting decisions are taken according to local parameters, this algorithm is able to get rid of the resolution limit.

*Lancichinetti et al.* introduced in [127] an algorithm aimed to unveil the hierarchical structure of networks, as well as the mesoscopic structure at different resolution levels. They define communities as the groups of nodes which maximize the following fitness function:

$$f_{\mathcal{G}} = \frac{k_{\text{in}}^{\mathcal{G}}}{(k_{\text{in}}^{\mathcal{G}} + k_{\text{out}}^{\mathcal{G}})^{\alpha}}, \tag{1.16}$$

where $k_{\text{in}}^{\mathcal{G}}$ and $k_{\text{out}}^{\mathcal{G}}$ are the total internal and external degrees of the nodes of module $\mathcal{G}$ and $\alpha$ is a positive real-valued parameter, which controls the size of the communities. The parameter $\alpha$ tunes the resolution of the method: small values of $\alpha$ lead to big communities, while large values of $\alpha$ yield small communities. The authors find that $\alpha = 1$ is a prudent choice, which also recovers the definition of weak community (see Eq. 1.2).

## 1.3 FROM SINGLE-LAYER NETWORKS TO MULTILAYER NETWORKS

We have shown in the previous chapter an approach to the mesoscopic analysis of complex networks by finding its underlying community structure. As stated, the problem of unraveling the communities of networks is aimed at finding cohesive groups of nodes based only on its relative connectivity. However, many times networks are tagged with categories that can differentiate nodes, or also different types of links that help to categorize groups beforehand, and then focus on the analysis of the different interaction patterns that emerge. This new level of knowledge (and complexity) has recently attracted the interest of the network science community.

The new paradigm of networks' structure is called *multilayer interconnected* networks, to differentiate from the consideration so far of single-layer networks. The next sections are devoted to the introduction of examples, nomenclature, descriptors and dynamical processes on top of multilayer interconnected networks. We will pay special effort on the description of a particular type of multilayer networks called multiplex networks, in which the nodes are replicated across layers representing the same entity but with different types on intra and interlayer connections. This latter topological representation is then used to assess the outcome of two coexisting spreading processes, which is explained in Chapter 3.

### 1.3.1 REAL WORLD IS INHERENTLY MULTILAYER

Real world systems are a great motivation for the need of representing systems in a multilayer manner. Real multilayer systems are not specific scenarios far from daily nature, instead an avid observer can find instances of such structures in multiple situations, for example in social networks. Previously, by means of the Zachary Karate Club network, we have seen that representing social interactions as a set of nodes accounting for people and a set of edges accounting for their friendship is enough to unveil some aspects of its structure, for example its division in communities. However, social networks contain more types of relations, such as familiar ties or work connections. If we wished to study the spreading of a rumor in a social network, taking into account these different categories would certainly lead to richer dynamics and we could observe additional effects. Indeed, in this particular example, the nature of the information received in the spread-

ing could alter the decision of the user to propagate it only to family, or only to people in his work environment.

A second example of an intrinsically multilevel system are transportation networks. We can represent the transportation system of a city by considering a set of locations which are connected if there is a way to go directly from one location to the other. However, in most cities, different modes of transportation coexist, for example, the bus, the tube or the car, and to provide a proper assessment on the quality of transportation, these different modes have to be taken into account. In a multilayer setup, the different transportation modes would be represented by edges of different categories, and by keeping this information separated we could address questions such as: "which is the optimum overlap between transportation modes to achieve the minimum amount of congestion possible?" or, "how does an interconnected network like this one respond to failures in certain locations?" [55].

Our third example comes from biology. The *Caenorhabditis elegans* is a small nematode which is famous for being the first living organism whose entire genome was sequenced. Indeed, biologists were even able to get a full mapping of the nematode neural network, in which the largest somatic nervous system (corresponding to the non-pharyngeal cells) consists in 282 neurons and approximately two thousand connections. The *C. elegans* neural structure has been widely studied in multiple fields, and in the complex networks environment it has been usually represented as a simple network containing only one level of description. However, it is known that neurons can be connected either by a chemical link or by an ionic channel, and that these two types of connections lead to completely different dynamics. Therefore, a more appropriate study of the structure of this nematode would require the representation of its structure in a multilayer network, connecting the 282 nodes with two differently categorized connections.

A vast amount of real situations can be modeled by using multilayer networks, either because their nodes are tagged according to some categories, or because the interactions between elements are different in nature, or both. In an effort to clarify the different types of multilayer structures and its different nomenclature, I offer my own classification of choice next.

### 1.3.2 TYPES OF MULTILAYER NETWORKS

The study of multilayer structures is relatively new to the complex networks field, gaining much attention in the past half decade. However, as we have seen in previous cases (e.g. the community structure of networks), these structures that we aim to study and characterize today were already the subject of attention of other scientists, mainly working in sociology-related fields. Indeed social networks are inherently multilayer, and sociologists recognized decades ago that it is crucial to study social systems by constructing multiple social networks using different types of ties among the same set of individuals [189, 216]. In this literature, the systems that we refer to have often been called *multiplex networks* [211] or *multi-relational networks* [216].

The rapid growth of this topic in complex networks, and the variety of approaches to the same problem have lead to a big amount of nomenclature, where sometimes the same structures are referred to with different names (and often the same name is used for different purposes), which is often confusing. Recently, in an effort to standardize definitions and tools for such networks, several reviews on the topic have been published (see [24, 122, 133]). The categorization of multilayer networks in different types is still lacking consensus. However, a plausible categorization that I find useful is the following (see Fig. 1.9 for a schematic representation):

a) *Node colored networks*: Here nodes are tagged according to different categories. Nodes may represent totally different entities or similar entities with additional considerations. Edges represent the same kind of connections along the network. An example of such structures is a power grid network formed by power stations and nodes in the Internet communication network [35], connected between them by different wires that all depend on electric power. Here, different colors represent different functions, as power station nodes provide electricity and Internet nodes are consumers of such energy.

b) *Edge colored networks*: In these networks, nodes belong to the same category but edges may have multiple natures. A simple example of this structure is a social network where the nodes are people and differently colored links represent different ties between people, e.g. family, friends, coworkers, etc.

c) *Node and edge colored networks*: Here, nodes represent different categories and also the connections between them are different. Is it easy to imagine this situation as two different networks that interact with each other in some way. To depict an example of such situation, and making use of the sketch of Fig. 1.9 (c), we could imagine that green nodes are companies, performing trading actions between them (represented by green edges); and pink nodes are banking entities, that also have interaction between them (pink edges). Between the two networks, further trading relations are allowed, such as loans offered by financial entities to companies (depicted by grey edges). These structures are also often called *interconnected* or *interdependent* networks.

d) *Multiplex networks*: This category represents a very particular situation. Here nodes are considered to be comparable entities, but the connections between them are of different kinds depending on the context. The two layers represented in Fig. 1.9 (d) have exactly the same amount of nodes, because they represent the same entities on a one-to-one correspondence (represented by dashed vertical lines). Nodes in different layers may have different states, and the interlayer link can be used to express coupling. An example of a situation well described by this setup is the spreading of information in two different online social platforms, e.g. Twitter and Facebook. If we take a set of individuals who are users of the two platforms, the differently colored links represent people they follow in Twitter and friends in Facebook. The inter-layer link can be used to model the delay there is between being an active spreader of a piece of information in Twitter and deciding it to spread it also in the Facebook network.

### 1.3.3 MATHEMATICAL DEFINITION OF MULTILAYER NETWORKS

Until now, we have always made use of the adjacency matrix $\mathbf{A}$ to represent a single-layer network. This matrix is square and of size $N \times N$ and its entries $a_{ij}$ define the connections between nodes, taking the value 1 if the connection exists and 0 otherwise. In a more general case, we have used the weighted adjacency matrix $\mathbf{W}$ whose elements $w_{ij}$ define the strengths or weights of the connections between nodes. To be able to exploit all the potential of multilayer networks, we need to find a mathematical object able to represent such a complex structure. It is only by finding a suitable mathematical definition that we will be then able

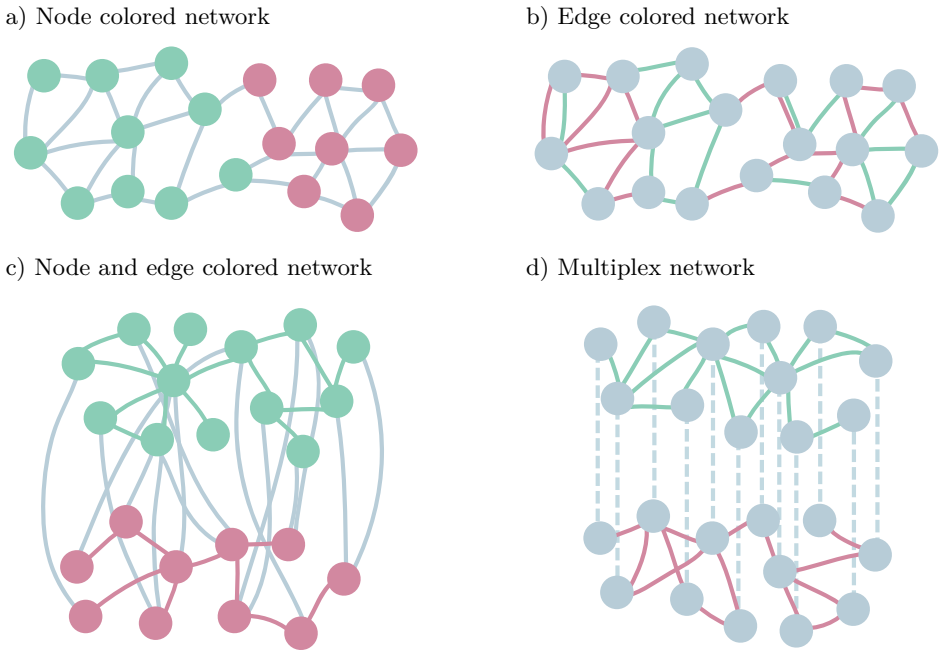



Figure 1.9: Schematic representation of the classes of multilayer networks, for the case where there are only two different types of nodes or connections.

to extend all our current knowledge of single-layer networks to the multilayer domain. We refer to multilayer networks, and we do not differentiate between the types of networks presented before, because we refer to the most general case (see Fig. 1.10).

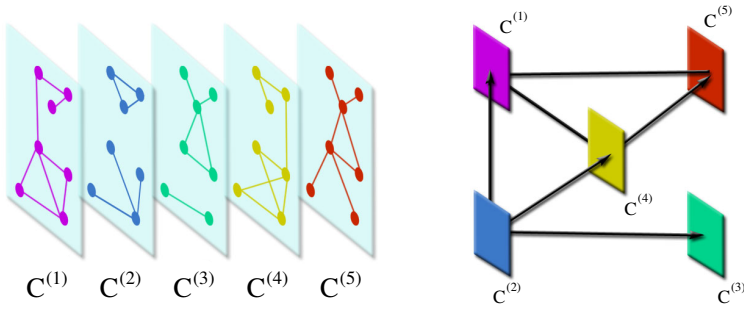



Figure 1.10: Schematic of a multilayer network. **Left.** Multilayer network consisting of five layers, only the internal connections are highlighted. **Right.** Connections between layers. This representation is schematic, therefore the arrow between layers can be understood as any type of connection between the nodes of both layers. *Original source: [52]. Reprinted with permission.*

Conveniently, multilayer networks are often represented by means of the *supra-adjacency matrix*. For a multiplex network with $N$ nodes and $L$ layers, the supra-adjacency matrix $\mathcal{A}$ is an $L \times L$ matrix of $N \times N$ blocks of layer-to-layer adjacency matrices. This takes the form:

$$\begin{pmatrix} A_{11} & A_{12} & \dots & A_{1L} \\ A_{21} & A_{22} & \dots & A_{2L} \\ \dots & \dots & \dots & \dots \\ A_{L1} & A_{L2} & \dots & A_{LL} \end{pmatrix}.$$

This matrix has, in its diagonal, the adjacency matrices for each layer (i.e. intralayer connectivity), and off–diagonal we have the matrices that represent the connections between layers (i.e. interlayer edges). This notation also supposes that all layers are equal in the number of nodes $N$, however if this is not the case, we might construct the single layer matrices taking into account the additional nodes but adding no connections to anyone else. By adding this isolated nodes we can construct then the square matrix we need. Also, if we account for weights either in the diagonal blocks or off–diagonal, we can weight the intralayer and

interlayer interactions, respectively. Equally, if we allow for non-symmetrical matrices we are defining a directed supra-adjacency matrix. It is worth noting that, by means of this representation, we are algebraically assuming that (I) all links have the same semantics —there is no formal difference between interlayer and intralayer links—, and (II) nodes in different layers represent different entities —which implies that the first node of a layer is not the same entity as the first node of the second layer, mathematically. This latter consequence implies that if we wish to make use of this representation to express the connectivity within a *multiplex network* —where nodes are replicated along layers—, we will have to formulate our measures so that we explicitly account for nodes being the same entity.

In the same way we defined the supra-adjacency matrix, we can also define the *supra-laplacian* $\mathcal{L}$ matrix of a multilayer network. The Laplacian matrix $\mathbf{L}$ of a single-layer network is defined as $\mathbf{L} = \mathbf{D} - \mathbf{A}$, where $\mathbf{D}$ is a $N \times N$ matrix with the degree of each node in the diagonal and zeros off–diagonal. The supra-laplacian matrix is then defined in the same way, but using the *supra-adjacency* matrix instead of $\mathbf{A}$.

An alternative form to express the previous supra-adjacency matrix is introduced in [52]. In brief, they make use of a tensorial object $\mathbf{M}$ with entries $M^{i\alpha}_{j\beta}$, which take the value of the connection between node $i$ in layer $\alpha$ and node $j$ in layer $\beta$. The benefits of encapsulating the representation of a multilayer in a fourth-order tensor is that it is a compact formulation, and there is a grounded mathematical framework to approach its treatment. Note that the previous single-layer representation is recovered by using a second-order tensor.

### 1.3.4 DESCRIPTORS OF MULTIPLEX NETWORKS

Plenty of literature of complex networks is devoted to the characterization of networks, using objective measures that aim to describe particular features of its structure. Some of those descriptors are the multiple definitions of centrality, the clustering coefficient and path distances, among many others. Once the multilayer concept is formally defined, the natural step is to extend the previous measures to the case of multilayer networks. However, this problem is far more complicated than the mere mathematical translation from a measure applied to a second-order tensor to a fourth-order tensor. More often than not, measures that have a clear definition in single-layer networks lose their semantics when translated to multi-

dimensional systems. As we will see, even something so intuitive and clearly defined in a single-layer network as the *degree of a node* becomes much fuzzier in the case of a multilayer network, and requires of the scientist's intervention and decision making. An extensive review of all the available measures for describing multilayer networks is out of the scope of this document (see the reviews [122, 24] for more details). However, to illustrate how the adaption of measures is done, and to show the change of semantics that the new definitions sometimes carry, in the following I will introduce some extended measures that are currently used to describe multiplex networks. We will focus on *multiplex* networks and not refer to multilayer in general because it is the structure that we will use in the contributions section.

#### 1.3.4.1 *Centrality and ranking of nodes*

One of the main problems in networks is to find which are those nodes that play a relevant structural role in the network. There is much information about how to identify such nodes in single-layer networks, and the most widely used approaches are based on degree, betweenness or eigenvector centrality. However, if we seek to find the most central nodes in a multiplex network, these measures have to be reconsidered. In a multiplex, the degree of a node $i$ would be defined as the vector: $\mathbf{k}_i = (k_i^{[1]}, k_i^{[2]}, \ldots, k_i^{[L]})$, where $L$ is the number of layers and each component of the vector is the degree of node $i$ in each layer [20, 18] (note that in the general multilayer case, interlayer degrees between any two layers $\alpha$ and $\beta$ $\{k_i^{[\alpha\beta]}\}$ need also to be specified). This definition of the degree is respectful of the multiplex structure, in the sense that it is not assuming layer aggregation, but it is cumbersome to work with. Measures of centrality are mainly used to obtain a ranking of the nodes according to some measure (in this case, the degree), but having to deal with degree vectors instead of scalars is a disadvantage. To produce a ranking, we would have to define an ordering, which would certainly lead to having to aggregate the vector in some way. Such aggregation can be produced simply by calculating the overlapping degree $\sum_{\alpha=1}^{L} k_i^{[\alpha]}$, or by using any other methods, such as a convex combination of $k_i^{[1]}, \ldots, k_i^{[L]}$, or any norm of $\mathbf{k}_i$ [24].

Similarly, approaches based on the spectral properties of the adjacency matrices, such as eigenvector centrality, can also be calculated for the case of multiplex and multilayer networks, as discussed in [2] and [195]. The simplest way to cal-

culate eigenvector centralities for the case of multiplex networks is to consider the eigenvector centrality in each layer separately, leading to another vector for every node: $\mathbf{c}_i = (c_i^{[1]}, c_i^{[2]}, \ldots, c_i^{[L]})$. As proposed in [195], the independent layer eigenvector-like centrality is then the matrix $C = (\mathbf{c}_1^T | \mathbf{c}_2^T | \ldots | \mathbf{c}_L^T) \in \mathbb{R}^{N \times L}$, where $T$ denotes the transpose. Other kinds of aggregations $f(\mathbf{c}_i)$ can be considered, such as the sum, the maximum or the $\ell_p$-norm.

A feature of the previous approaches is that they calculate the centrality of nodes for each layer separately, thus not considering the interactions between layers. A measure that takes into account such interdependency is presented in [53], where the authors propose the calculation of the leading eigentensor $\Theta_{i\alpha}$ of the tensorial object $M_{j\beta}^{i\alpha}$, as the solution of the tensorial equation $M_{j\beta}^{i\alpha}\Theta_{i\alpha} = \lambda_1 \Theta_{j\beta}$, with $\lambda_1$ being the largest eigenvalue of $\mathbf{M}$. Here, $\Theta_{i\alpha}$ encodes the eigenvector *versatility* of each node $i$ in each layer $\alpha$, when accounting for the whole interconnected structure. The resulting matrix has to be aggregated by layers if we wish to obtain a ranking of the nodes, as in the previous methods, but the main novelty of this approach is that here we respect the multiplex structure, instead of treating it as a collection of single-layers. Furthermore, another interesting consideration that this work makes is that here, the concept of centrality on networks is redefined according to multiplex structures. The authors refer to their formulation as a way to obtain the *versatility* of a node, in which nodes will obtain high versatility score if they are central in the whole structure of the multiplex.

#### 1.3.4.2 *Correlations between layers*

An important feature to take into account when characterizing multiplex networks is that, in addition to the correlation properties of individual layers, one can also study the correlation *between* layers. In other words, we are interested in the correlation of the degrees of a node across layers. The idea that degrees of nodes may show correlation across layers is intuitive, e.g. in a multiplexed social network a node with high degree in a friendship environment is very likely to define a social person, and therefore this node is likely to be a hub also in a work or familiar environment. This can or cannot be the case, therefore studying the topological correlations of the different layers of a multiplex is a good way to a better understanding of the structure we are dealing with. Furthermore, in the presence of interlayer degree correlations, the joint degree distribution $P(\mathbf{k})$ does not factorize into the product of individual layer's degree distribution.

To quantify the interlayer degree correlation, some measures based on the correlation coefficients were introduced. An example of such a measure is the Pearson correlation coefficient between two layers $\alpha$ and $\beta$,

$$\rho_{\alpha\beta} = \frac{\langle k_\alpha k_\beta \rangle - \langle k_\alpha \rangle \langle k_\beta \rangle}{\sigma_{k_\alpha} \sigma_{k_\beta}}, \tag{1.17}$$

where $k_\alpha$ denotes the degrees of the same nodes in different layers, and $\sigma$ refers to the standard deviation. Other correlation coefficients like Spearman or Kendall's have also been considered [162].

To investigate the dynamic implications of interlayer degree correlation, it is usual to compare three specific patterns of correlated coupling, which are the maximally-positive coupling (MP), the maximally-negative coupling (MN) and the random (uncorrelated) coupling [132]. Given two layers, the MP coupling is build by connecting high degree nodes from one layer to the high degree nodes of the other layer, and equally with the low-degree ones, thus giving an assortative interlayer connectivity. Its counterpart, the MN coupling is achieved by doing the opposite, which means connecting nodes with very dissimilar degrees. The uncorrelated version of such coupling is achieved by randomly connecting nodes from different layers, without any consideration on their degree, see Fig. 1.11 for an illustration of the resulting networks.

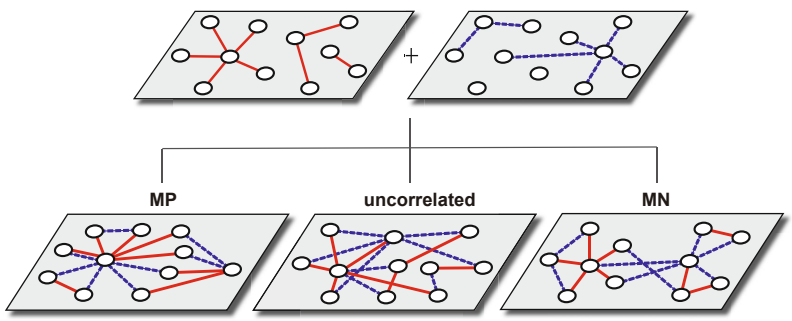



Figure 1.11: Schematic illustration of three patterns of interlayer degree-correlated multiplex networks: maximally-positive correlated, maximally-negative correlated and uncorrelated. *Original source: [133]. Reprinted with permission.*

Another interesting measure related to the correlation between layers is the existence of link overlap across different layers. Indeed, this is studied in social

network literature, by asking how the presence of a link between two nodes in a certain layer facilitates or hampers the existence of the same link in another layer. From the theoretical point of view, assuming sparse networks, the link overlap would be unlikely to exist if the layers were coupled completely at random. Therefore the existence of such overlap would reveal underlying non-randomness of layer coupling in the system.

#### 1.3.4.3 *Clustering coefficients*

In single-layer networks, clustering coefficients can be calculated either as local or as global measures. As a local measure, the clustering of a node $i$ is usually defined as the fraction of actual edges between the neighbors of $i$ with respect to the maximum possible number of links between the neighbors of $i$. On the other hand, the global definition accounts for the number of triangles in the network divided by the total number of triads (i.e. sets of three nodes connected with at least two edges) [130], or sometimes it is calculated as the average of the clustering coefficient of all nodes. Plenty of definitions of clustering in single-layer networks coexist, and the dimensions added in the multilayer and multiplex scenarios cause the amount of possibilities to rise. The added complexity is that now, the concept of what is a neighbor and what is a triangle are lost. In a multiplex scenario, we could be interested in considering that the neighborhood of a node $i$ accounts for all the nodes $j$ that are connected to $i$ in any layer, or we might want to consider layers separately, or a consider only a subset of them. Similarly, when defining what is a triangle, we might count also the interlayer links, or consider only the contribution of those triangles that connect nodes from different layers (see Fig. 1.12 for an illustration on the possible ways to create a triangle considering multiple layers). In this new scenario the calculation of a clustering coefficient will require a tailored design of such measure, to accommodate the particular needs of the feature we want to evaluate. Some clustering coefficient formulations for multilayer and multiplex networks can be found in [47, 15, 32, 48, 18].

#### 1.3.4.4 *Paths and distances*

The concepts of paths in a network, the associated distances and walks are important in both graph theory and network science. In brief, paths are sequences of nodes that have to be crossed to go from an origin node to a destination node, where the intermediate nodes can only be visited once. The length of the path

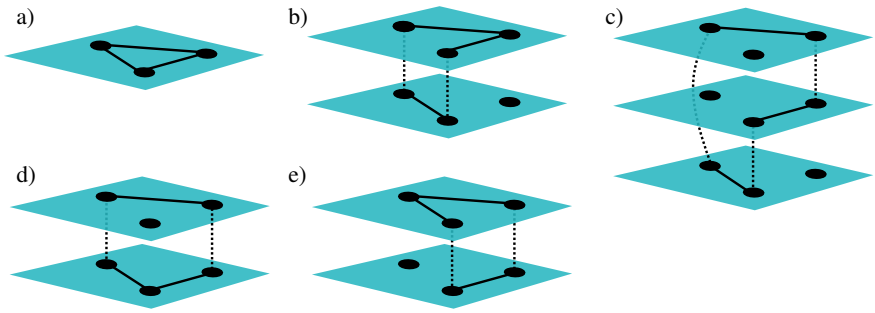



Figure 1.12: Sketch of five possible combinations of links of different layers to form triangles. *Modified picture from original source: [47]. Adapted with permission.*

connecting these two nodes is the number of edges that had to be traversed in the path to get to the destination node. If the graph is weighted, and edges between nodes somehow refer to a cost, length calculation must be adjusted to account for such heterogeneity. These measures are important because they are a basic keystone to calculate other measures, such as graph distance, connected components, betweenness centralities and random walks, among others. These tools can also be used to define additional methods based on them, such as community detection algorithms or centrality measures. Therefore, the translation of the concepts of paths and distances to the case of multiplex networks is crucial. However, as we noted in the case of clustering, the increased amount of dimensions makes the definition of a path more complicated, where one of the decisions that has to be taken is if we count interlayer links or not. The answer to this question often depends on the particular setup we are representing. For instance, if our multiplex network accounts for different modes of transportation (e.g. subway, bus, train), then the interlayer links account for transferring, in the same location, between two of these modes, and it is plausible to think that the associated cost of this operation is not negligible. In this case, it is often natural then to generalize the concepts of paths from single-layer networks by simply replacing nodes with node-layer tuples. This approach has been used to generalize concepts like random walks [54]. On the other hand, if our system represents a social network consisting of multiple layers, where each node is an individual, then, depending on the associated dynamics, it may not be necessary to account for interlayer links, which effectively means that when considering measures based

on distances and walks, jumps between layers will not contribute to the cost (see Fig. 1.13 for an illustration of such concept).

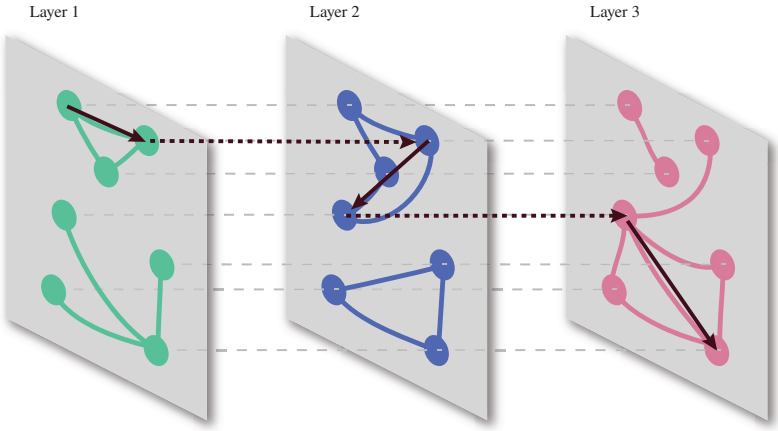



Figure 1.13: Example of a path between two nodes in a three layer multiplex. Dashed arrows represent inter-layer jumps and solid arrows account for intra-layer steps. The length of this path would be 5 or 3, depending if we count inter-layer edges or not, respectively. *Custom-made picture inspired from the original in: [55].*

### 1.3.5 DYNAMICS ON MULTIPLEX NETWORKS

Aside from the characterization of the topological structure of multiplex networks, which leads to new interesting measures and definitions, a very important subject is the analysis of dynamical processes on top of such networks. The outcome of dynamical processes on networks has a strong dependence on the underlying network topology, namely the type of networks that conform each layer and the connectivity between them. And often the effect induced by accounting for multiple layers is non-additive and non-linear in the individual layer's effect. As introduced previously, interlayer connections can be defined in multiple ways, mainly conforming two types of multiplexes, those where the layers are coupled in a cooperative manner, and those that have a complementary coupling. The first case comprises situations where the correct functioning of one layer is dependent on the proper functioning of another layer, as it happens with the case of interdependent networks [35]. It also accounts for processes in which a node's state is affected by synergistic influence from the states of multiple layers, such

as multiplex threshold cascade [34, 131]. On the other hand, complementary-coupled multiplexes, like for instance, transportation systems, are designed in such a way in which a layer provides alternative paths between the same set of nodes via other layers. In such systems, malfunction of one single layer may not fundamentally alter the functionality of the whole system.

Either way, dynamical processes on top of multiplex networks exhibit the emergence of new effects, that are not observed when considering single-layer networks. One paradigmatic example of how the structure is able to influence the dynamics is the case of diffusion. Diffusion is one of the simplest dynamical processes, which has also been the object of many studies in single-layer networks [109, 163], finding that the key determinant of diffusion dynamics on networks is the spectral structure of the Laplacian matrix. For the case of multiplex networks, a mathematical formalism that makes use of the supra-Laplacian matrix was developed in [91, 196]. In these works, authors start from a multiplex network of $L$ layers and $N$ nodes in each layer, in which particles diffuse with diffusion constant $D_\alpha$ in layer $\alpha$ and with constant $D_{\alpha\beta}$ when diffusing across different layers. This last term can also be referred to as the coupling parameter. The time evolution of particle densities at each node and each layer are determined by:

$$\frac{dx_i^{(\alpha)}}{dt} = D_\alpha \sum_{j=1}^{N} w_{ij}^{(\alpha)}(x_j^{(\alpha)} - x_i^{\alpha}) + \sum_{\substack{b=1 \\ b\neq a}}^{L} D_{\alpha\beta}(x_i^{\beta} - x_i^{\alpha}), \tag{1.18}$$

where $w_{ij}^{\alpha}$ is the link-weight matrix of layer $\alpha$ which, in matrix form, is expressed as:

$$\frac{d\mathbf{x}}{dt} = -\mathcal{L}\mathbf{x}, \tag{1.19}$$

where $\mathbf{x} = (x_1^{(1)}, \ldots, x_N^{(1)}, \ldots, x_1^{(L)}, \ldots, x_N^{(L)})^T$ and $\mathcal{L}$ is the supra-Laplacian matrix of the multiplex. A parameter of special importance is the smallest nonzero eigenvalue $\lambda_2$ of the supra-Laplacian matrix, which sets the diffusion timescale $\tau$ to $\tau = 1/\lambda_2$, which characterizes how fast the system can relax to the stationary state. The main conclusions from [91] for the case where we have two layers, are that the diffusion timescale $\tau$ undergoes a qualitative change at certain threshold value of the interlayer diffusion strength $D_{\alpha\beta}$. For low values of the diffusion coefficient between layers, the diffusion time scale of the global system is controlled by the inverse of $2D_{\alpha\beta}$. Oppositely, if $D_{\alpha\beta}$ is above the threshold, the timescale becomes dependent on the details of the multiplex coupling, and the diffusion on

the multiplex is always faster than the diffusion in the slowest layer of the two. It is only for sufficiently large values of $D_{\alpha\beta}$ and for certain configurations of the topology that the diffusion on the multiplex can become faster than the diffusion on any of the layers in isolation, phenomena which the authors call *superdiffusion*.

Another dynamical process that has been studied on multiplex networks is the percolation process, which was also studied before in single-layer networks [63, 40, 39, 212]. Percolation is a classical problem concerning the global connectivity of a system, thus its study in networks consisting of multiple layers needs of a generalized notion of connectivity. A mutually-connected component (or mutual component for short) [35, 197] is defined as the set of nodes in which each pair is connected within each and every layer simultaneously. The biggest mutual component is then called the giant mutual component, and it is the order parameter of mutual percolation studies. A key finding was presented in [35] regarding cascades of failures in interconnected networks. In that work, the authors found a discontinuous transition in the size of the giant mutual component at the critical fraction of random node removals, which they called an abrupt collapse of the system. Those results were enlightening and encouraging, given that such a transition was clearly induced by the underlying network topology. After that, the problem was re-addressed in purely structural terms in [197], allowing an accessible analytical approach [19] aimed to extend single-layer percolation formulation to the case of multiplex networks.

A problem related to percolation that also has been extensively studied in single-layer networks is that of robustness of networks against attacks [43]. Network robustness considers the effect of the removal of nodes when this is not done randomly, as it happens in percolation problems. Instead, it focuses on *targeted* or intentional attacks [5, 42], mainly by removing nodes according to their degrees. The main quantity used to assess the robustness of a network after the removal of nodes is the giant mutual component, which largely depends on the correlation between the two layers. In the case of random failures, positive (assortative) interlayer degree correlation is shown to enhance the robustness of the mutual connectivity in multiplex random networks. Oppositely, negative (disassortative) interlayer degree correlation causes the robustness to diminish. In the case of targeted attacks, the robustness then does not only depend on the interlayer correlations, but also on the initial density of links in all layers [144].

Games have also been approached from the perspective of multiplex networks. For the case of single-layer networks, the study of evolutionary games has at-

tracted a lot of interest as a connector between statistical physics, evolutionary dynamics and social sciences [171]. Previous research mainly focused on social dilemmas, in which agents can choose between two strategies: *cooperate* or *defect*. The most famous and studied social dilemma is the *Prisoner's Dilemma Game*, which addresses the emergence of cooperation [12], and its $n$-player counterpart, the *Public Goods Game* [117]. Later, such models were studied in complex networks [185], until the present time, where the generalization to interconnected and multilayer networks was introduced [94, 113, 140]. As an example of application of games to multiplex networks, the authors in [94] considered a two-layer setup, in which each node $i$ can in principle take different strategies in each of the layers, but the payoff is computed globally and is accessible to neighbors of node $i$ in all layers.

Aside from the previous processes, one can also study the effect of a multiplex topology on epidemic spreading processes. Epidemic spreading is a wide topic that will be introduced appropriately in Section 1.4. In short, epidemic spreading aims to model how infectious processes (diseases, information, etc) spread through a network of contacts. Epidemics on top of single-layer networks have been extensively studied [168]. After, SIS and SIR models (see Sec. 1.4) were introduced for interconnected networks [214, 187, 60]. Multiplex networks can be used to model epidemics that spread through different channels. In this case, it was shown that the epidemic threshold in such systems is completely governed by the layer with the largest maximum eigenvalue of the contact probability matrix, and that the process could not be correctly described by means of an aggregated network [46]. Also, multiplex networks can be used to model spreading processes where there is some kind of spatiotemporal separation between two spreading processes, e.g. online and offline communication channels for information spreading, where the layer-switching cost accounts for this separation [37]. The case where the multiplex representation accounts for a network that evolves in time has been studied in [209]. Another situation for which the multiplex framework is useful is to assess the outcome of the spreading of an epidemic using different topologies. Particularly interesting is the case where the two spreading processes interact between each other in some way, either by collaborating or by hampering each other's spreading. One example could be the spreading of two different pathogens that transmit via different mechanisms, such as HIV and Tuberculosis [186], or spreading of competing memes [49]. Another exciting problem is that of the interplay between awareness and disease spreading, a problem addressed by who writes these lines, which will be explained thoroughly in Section 3.

## 1.4 EPIDEMIC SPREADING PROCESSES ON COMPLEX NETWORKS

One of the reasons why scientists working in complex networks have great interest on epidemics is because of the natural suitability of networks to represent epidemic processes. Diseases spread between people in different ways: air-transmitted diseases such as tuberculosis or influenza spread when two individuals breath close to each other, sexually-transmitted diseases spread when two people have intercourse, and contagious parasites are transmitted if individuals touch each other. All these transmissibility patterns can be modeled as networks, and used as a substratum of epidemic spreading models [151].

Actually, the biological process that takes place when an individual gets infected is very complicated. The pathogen multiplies in the individual's body while the immune system tries to hold it back, resulting in a biological fight that might ultimately lead to the individual's recovery, death, or a state of chronic infection. If we are interested in understanding how a disease spreads through the population, in theory we should take into account all the previous biology, but this would lead to an overwhelming amount of parameters that would make our system mathematically intractable in practice. Also, for obvious reasons, experimenting epidemics *in vivo* is not a viable option, therefore in the past modeling approaches have been the best option to learn about these processes. These approaches, based on simple models of the spreading of diseases are useful to assess the final outcome of epidemic processes, test theories and design intervention strategies.

Mathematical models use the language of mathematics to describe a system. In epidemiology, models allow us to translate between behavior at various scales, or to extrapolate from a known set of conditions to another. Models allow us to predict the population-level epidemic dynamics from an individual-level knowledge of epidemiological factors, the long term behavior from the early invasion dynamics, or the impact of vaccination in the final outcome of the infection. In general, models have two possible roles, mainly *prediction* and *understanding*, although in the majority of cases it is difficult to find a model where these two roles coexist. A predictive model requires the maximum accuracy possible, and therefore needs the inclusion of all the known complexities and population-level heterogeneities. Predictive models are of great use in specific situations, such as in a real epidemic outbreak, being useful to guide policy-makers and institu-

tions on applying controlling measures on the population. On the other hand, models can also be used to understand how an infectious disease spreads, and how various complexities affect the dynamics. In essence, these models provide epidemiologists with an ideal world in which individual factors can be examined in isolation and where every facet of the disease spreading is recorded in perfect detail. Although it might seem that such *toy models* are driven purely by scientific curiosity and have little to do with the real implications of the course of an epidemic, the knowledge gained by studying such models is often robust and generic, and therefore can be applied to a wide variety of particular problems. Besides, such approaches allow scientists to build an intuition for infection patterns, a necessary step before approaching epidemics with more complicated models. In this section, we will focus on this latter approach, starting out from the classical most simple epidemic models and their accompanying mathematical formulations.

Interestingly, models for epidemic spreading are also able to describe a wide variety of phenomena besides the spreading of infectious diseases among humans and animals. Viruses spreading between computers through the Internet, the spreading of information in social networks or the adoption of technology or cultural norms are all processes that can be mathematically described by models of contagion processes. Indeed, even though the nature of such phenomena and their transmission mechanisms are different, all of them present similar dynamical behaviors, which allows a similar description in terms of epidemic processes. This multi-purpose nature of epidemic processes makes epidemic spreading modeling a highly multidisciplinary topic, with a variety of different approaches and models, ranging from the most simple explanatory, to the most realistic and intricate ones.

In the past years, epidemic modeling has revived a second golden age (see a review in [168]). The recent availability of large-scale data sets together with the increase in computational power has allowed the explicit simulation of entire populations down to the scale of single individuals. Indeed, mobile and wifi technologies are used regularly in our daily life, and supply a huge amount of information about user-to-user interactions for millions of individuals at once. Also, online social networks represent a source of traces that individuals leave in their daily activities. All this sources allow the measurement of interesting patterns of behavior of the users, which ultimately facilitate the simulation of spreading of information, opinions, habits, etc. These tools have helped the evolution of mathematical models into models that are able to simulate epidemics

at an individual level, which has been very useful, in particular in the context of infectious diseases, providing crucial information used in policy-making [210, 174].

Due to the increased interest in epidemic modeling, there is plenty of literature that approaches the subject from diverse points of view [7, 8, 103, 119, 31, 61]. In the following, I will introduce some of the classical mathematical models used for epidemic spreading processes, digging deeper in those models used in the contributions section of this document and therefore necessary for a clear understanding. Although the terminology used will most of the times refer to the context of infectious diseases, the models presented are suitable, as mentioned, for most social and information contagion processes that fit into the epidemics metaphor.

### 1.4.1 TRADITIONAL MODELS FOR EPIDEMIC SPREADING IN HOMOGENEOUS POPULATIONS

In this section I will introduce the basic building blocks for most epidemiological models: the compartmental SIS and SIR models. For these models it is possible to develop some analytical results, which are useful in the understanding of simple epidemics and in our interpretation of more complex scenarios. At this point we will focus on *homogeneous populations*, which means that we will neglect any kind of heterogeneity in the population (therefore ignoring age, gender or behavior patterns). Note that by neglecting the behavior patterns we are ignoring the connectivity between individuals, i.e. the underlying network of contacts, which means we assume that any individual can be in contact with any other individual in the network. We will also consider that the population is stable in number, therefore we do not consider mortality or birth rates, nor do we consider migration. This simplification will allow us to introduce the classical mean-field mathematical formulation for the SIS and SIR models, which is a necessary first step towards developing models that do take into account heterogeneities, which we will introduce later in Sec. 1.4.2.

#### 1.4.1.1 *Stages of an infectious process*

Traditional modeling approaches make use of compartments that describe the stages of an individual during the course of an infection. The main commonly

used stages are the following: at first, individuals (or *hosts*[6]) are in the *susceptible* (S) state, which mean they are healthy but can contract the infection. This susceptible individual may encounter an infectious agent and become infected with a pathogen. The amount of pathogen in the host's body multiplies over time, but in early stages this quantity may be still too low to allow further transmission. Individuals in this phase are said to be in the *exposed* (E) class. Once the level of pathogen is sufficiently large within the host, it can be potentially transmitted to other individuals, which causes the individual to be in the *infectious* or *infected* (I) class. Finally, if the host's immune system is able to fight out the pathogen, the individual is said to be *recovered* or *removed* (R), accounting for individuals that are no longer infectious due to recovery from the illness or due to death, respectively.

This fundamental classification in *susceptible*, *exposed*, *infectious* and *recovered* stages only depends on the host's ability to transmit the pathogen, making irrelevant the disease status of the host. Examples of diseases whose process follows strictly these four stages are measles, rubella or chickenpox. However, in other kinds of disease, it is often justifiable at the population scale and mathematically simpler to ignore the exposed class, reducing the number of equations by one and leading to the SIR (susceptible-infectious-recovered) dynamics. Alternatively, some infections are better described by SI (susceptible-infectious) dynamics, such as some infections in plants, where the host is infectious soon after contacting the pathogen (thus the exposed class can be ignored) and it remains infectious until its death. Finally, some diseases (for example sexually transmitted diseases) are naturally suited to the SIS (susceptible-infectious-susceptible) paradigm, where the host is infectious for a period of time, but after recovery it can contract the disease again, leading to cyclic dynamics. Finally, SIRS dynamics represent those cases in which the host is immune after recovery for a period of time, and after that it becomes susceptible to infection again.

Aside from these classical models, additional compartments can be added to accommodate other possible states of individuals with respect to an infection, for instance immune individuals. In principle, the scientist is able to build a model as complicated as wanted by adding more compartments and modeling the transitions between them. These transitions, which are easy to understand as a verbal argument, must be translated into formal mathematical terms, in order to be able to make quantitative predictions on the final outcome of the

6 *Host* is the word used by epidemiologists to design individuals in the epidemic dynamics. In this document I will use the term *host* or *individual* interchangeably.

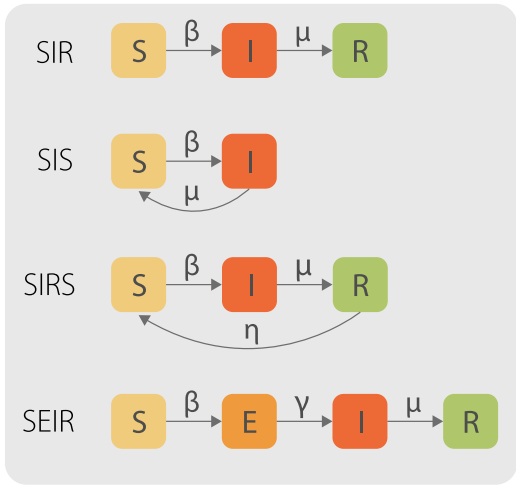



Figure 1.14: Diagrammatic representation of different epidemic models in terms of reaction-diffusion processes. Boxes stand for different compartments, while the arrows represent transitions between compartments according to their respective rates. *Original source: [168]. Reprinted with permission.*

models (see Fig. 1.14 for a diagrammatic representation of the mentioned models in terms of reaction-diffusion processes). Next, I will present the mathematical formulation for the SIS and SIR models, probably the two most commonly used paradigms, as well as introducing the most interesting quantities that we can obtain from solving the equations governing these models.

#### 1.4.1.2 *Formulation for the deterministic SIR model*

The Susceptible-Infectious-Recovered (SIR) model, initially studied by [120], assumes that the disease confers immunity once the infectious period has passed. This model has two transitions: S $\rightarrow$ I and I $\rightarrow$ R. The first happens when a susceptible individual interacts with an infectious individual and becomes infected. The second transition occurs when an infectious individual recovers from the disease and is assumed to have acquired a permanent immunity, or has died. This latter transition is spontaneous and does not depend on the state of any

other individuals in the population. The reaction-diffusion equations for these transitions are:

$$S + I \xrightarrow{\beta} I + I$$
$$I \xrightarrow{\mu} R$$

In the long time regime, the number of infected individuals always tend to zero, being absorbed by the R state.

To introduce the model equations, we will consider a closed population without demographics (the number of individuals is stable and additionally we do not consider births, deaths or migration). The scenario we will represent is a large population into which a low level of infectious agent is introduced and where the resulting epidemic occurs sufficiently quickly so that demographic processes are not influential. We also assume homogeneous mixing. Given the premise that underlying epidemiological probabilities are constant, the equations for the SIR model are as follows:

$$\frac{dS}{dt} = -\beta SI \tag{1.20}$$
$$\frac{dI}{dt} = \beta SI - \mu I \tag{1.21}$$
$$\frac{dR}{dt} = \mu I \tag{1.22}$$

The quantities $S$, $I$ and $R$ refer to the *proportion* of individuals in the susceptible, infectious and recovered states, respectively, and obey the normalization condition $S + I + R = 1$. The parameter $\mu$ is called the removal or recovery rate, where its reciprocal $1/\mu$ determines the average infectious period (in a continuous-time formulation and assuming a Poisson process) [45]. The parameter $\beta$ accounts for the probability of contagion given that a contact between an infectious host and a susceptible individual is produced. We can also define the *force of infection* $\alpha$, which expresses the probability at which one susceptible individual may contract the infection in a single time step. In the continuous-time limit it is defined as $\alpha = \beta \frac{N^I}{N}$, where $N^I$ is the total number of infectious individuals in the population, therefore following the notation used in Eqs. 1.20-1.22 we can also write it like this: $\alpha = \beta I$. These equations have the initial conditions $S(0) > 0$, $I(0) > 0$ and $R(0) = 0$.

Note that we are using the *frequency dependent* (or mass action) formulation, which considers that the number of contacts is independent of the population size;

and *not* the *density dependent* formulation, which assumes that as the population size increases, so does the contact rate[7].

Despite its extreme simplicity, the system of Eqs. 1.20-1.22 cannot be solved explicitly. That is, we cannot obtain an exact analytical expression for the dynamics of $S$, $I$ and $R$ through time, instead the model has to be solved numerically. Nevertheless, from these equations we can derive an interesting quantity, which is the *epidemic threshold*, defined as the relation between $\beta$ and $\mu$ that allows the disease to spread.

To calculate the epidemic threshold, we will consider the initial stages after $I(0)$ infectives are introduced in a population consisting of $S(0)$ susceptibles. To find the relation between $\mu$ and $\beta$ that will allow or not this disease to spread through the population, we need to rewrite Eq. 1.21 in the form:

$$\frac{dI}{dt} = I(\beta S - \mu). \tag{1.23}$$

In order for the disease to die out, we ask for $\frac{dI}{dt} < 0$, leading to $S < \mu/\beta$, since $I$ cannot be negative. In other words, if the initial fraction of susceptible individuals is less than $\mu/\beta$ then the infection *dies out* [120]. This is referred to as the *threshold phenomenon*, and establishes that there exists a proportion of susceptible individuals that we must exceed in order for the infection to invade. The inverse of this quantity is called the *basic reproductive ratio* (represented by the symbol $R_0$ and also called *effective spreading rate* in the networks jargon), and is one of the most important quantities in epidemiology. It is defined as the average number of secondary cases arising from an average primary case in an entirely susceptible population [61], and essentially measures the maximum reproductive potential for an infectious disease. Using this quantity to express the threshold phenomenon, we can say that the infection can invade in an initially susceptible population only if $R_0 = \beta/\mu > 1$. It is a very intuitive statement, as an infection that, on average, cannot infect more than one new host is not going to be able to spread significantly in the population [136].

Another interesting finding that can be extracted from the previous equations is the possibility to numerically approximate the curve of the fraction of infectious

7 The rationale behind this latter approach is that if individuals are crowded within a given area (and move randomly), then the contact rate will be increased. Experimental studies on the estimates of measles transmission rates in England and Wales demonstrate no relationship with population size [22]. We will not use the density dependent formulation and will focus on the frequency dependent only.

individuals as a function of $R_0$ [119]. We start out by dividing Eq. 1.20 by Eq. 1.22, and obtain:

$$\frac{dS}{dR} = -\frac{\beta S}{\mu} = -R_0 S. \tag{1.24}$$

Integrating with respect to R, we obtain:

$$S(t) = S(0)e^{-R(t)R_0} \tag{1.25}$$

Given that $e^{-R(t)R_0}$ is always positive, and $R \leq 1$, $S$ must remain above $e^{-R_0}$, which means that there will be always some susceptibles in the population who escape infection. This leads to a nice conclusion: the chain of transmission eventually breaks due to the decline in infectives, not due to a complete lack of susceptible individuals [119]. Taking into account that $S + I + R = 1$ and that the epidemic ends when $I = 0$, we can re-write the long term-behavior of Eq. 1.25 as:

$$S(\infty) = 1 - R(\infty) = S(0)e^{-R(\infty)R_0} \Rightarrow 1 - R(\infty) - S(0)e^{-R(\infty)R_0} = 0, \tag{1.26}$$

where $R(\infty)$ is the final proportion of recovered individuals, which is also equal to the proportion of the population that at some point was in the infectious state. An approximation of this equation can be made, and therefore we are able to plot the number of infected individuals as a function of $R_0$, as shown in Fig. 1.15. As expected, if $R_0 < 1$, then the epidemic dies out. On the contrary, if the basic reproductive ratio is large enough, the disease spreads through a good amount of the population (e.g. if $R_0 = 5$, more than the 99% of the individuals in a well-mixed population will contract the disease). Note that it is not difficult to have real cases in which $R_0$ has large values: estimated values of $R_0$ in real data of infectious diseases in humans state that influenza has $R_0 \simeq 3$ or 4 [147], rubella has $R_0 \simeq 6$ or 7 [7], while measles and whooping cough have $R_0 \simeq 16$ to 18 [6].

As mentioned, the exact solution of the equations of the SIR model is not feasible due to the nonlinear transmission term $\beta SI$. However, we can obtain an approximation of the *epidemic curve*, which is defined as the number of new infections per time interval [213, 104]. The approximation obtained by exploring the equation involving $\frac{dR}{dt}$ (see the complete calculation in [119]) is:

$$\frac{dR}{dt} = \frac{\mu\alpha^2}{2S(0)R_0^2}\text{sech}^2\left(\frac{1}{2}\alpha\mu t - \phi\right). \tag{1.27}$$

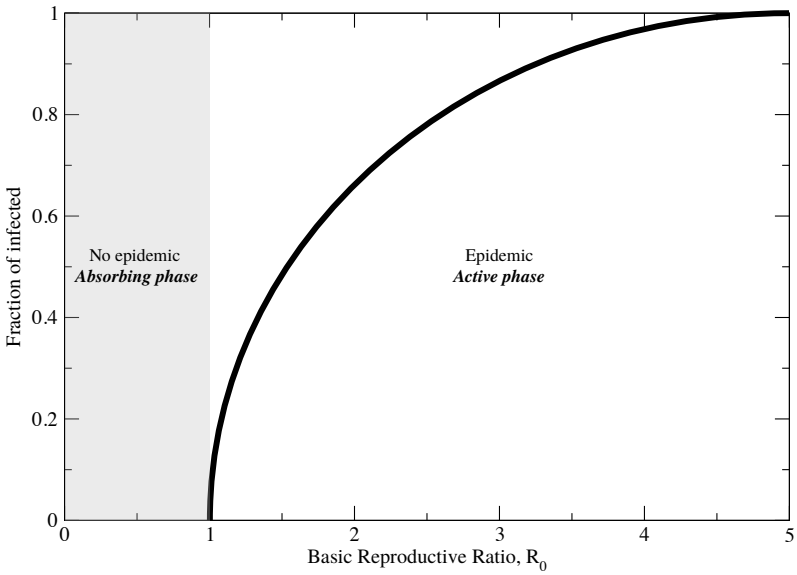



Figure 1.15: Total fraction of the population infected as a function of the basic reproductive rate $R_0$. This curve is obtained by applying the Newton-Raphson method on Eq. 1.26. *Source: custom-made figure resembling the one in: [119].*

This equation approximates the number of recovered individuals as a function of time. However, an important assumption was made during the derivation of this result. A necessary step was to consider that the factor $R_0 R$ was small. This will be true at the start of the epidemic, because still there are not many recovered individuals, or if the epidemic has a low $R_0$. This assumption translates in obtaining not very accurate results for diseases that are highly infectious. Instead, a common approach is to solve the set of Eqs. 1.20-1.22 numerically.

#### 1.4.1.3 *Formulation for the deterministic SIS model*

The previous SIR model accounts for infections that confer lifelong immunity once the subject is recovered from the infectious state. However, numerous diseases do not provide with immunity, such as sexually transmitted infections, rotaviruses and many bacterial infections. For these diseases, a subject can be infected multiple times during its lifetime. The SIS model accounts for such infectious diseases, where the subject returns to the susceptible compartment after recovery. The transitions between stages are then S $\rightarrow$ I and I $\rightarrow$ S, governed by the same probabilities as before, $\beta$ and $\mu$, respectively. The reaction-diffusion representation of this model is:

$$S + I \xrightarrow{\beta} I + I$$
$$I \xrightarrow{\mu} S$$

Supposing a well-mixed population and no demography, the SIS model is then described by a pair of ordinary differential equations:

$$\frac{dS}{dt} = \mu I - \beta I S \tag{1.28}$$
$$\frac{dI}{dt} = \beta S I - \mu I. \tag{1.29}$$

These equations fulfil $S + I = 1$. As in the case of SIR dynamics, the basic reproductive ratio is $R_0 = \beta/\mu$, and the equilibrium state will be stable as long as $R_0 > 1$. The SIS dynamics allow for a finite fraction of infected individuals in the stationary state, however convergence to the equilibrium is monotonic with no oscillatory behavior, as opposed of SIR.

### 1.4.2 TRADITIONAL MODELS FOR EPIDEMIC SPREADING IN HETEROGENEOUS POPULATIONS

The previous models compartmentalize the population according to the stages of the disease, and model the evolution of the number of individuals in each compartment. By assuming there is only one degree of subdivision within the population, they assume everybody behaves equally, a useful simplification that in some cases can lead to inaccurate results. Some infectious diseases are better modeled by introducing a second subdivision in the population, according to individuals with similar behavioral characteristics. Childhood diseases (e.g. measles, mumps or chickenpox) are most commonly suffered by young aged individuals, and therefore its modeling should take into account the age division of the population. Similarly, sexually transmitted diseases are better modelled by considering the behavior patterns (i.e. number of sexual partners) of individuals. Models that wish to account for heterogeneities in the population should choose a division in the population according to some characteristics so that all members of these new classes have comparable risk of both contacting and transmitting the infection.

To do this, a first approach would be to divide the population in two classes: *high–risk* and *low–risk*. Individuals belonging to the high–risk compartment would be, using the previous examples, young children in the case of modeling measles or sexually promiscuous people in the case of modeling HIV. We assume that individuals cannot change from one compartment to another, and also that the recovery occurs at a constant rate $\mu$ equal for all the population. We would then denote the fraction of infectious and susceptible individuals in the high–risk group as $I_H$ and $S_H$, respectively, and $I_L$ and $S_L$ for the low–risk compartment, with $n_H$ and $n_L$ being the total number of individuals in each group. Given this division of the population, we assume that there are different transmission patterns between members of the two groups, leading to a *matrix* of transmission parameters like the following:

$$\beta = \begin{pmatrix} \beta_{HH} & \beta_{HL} \\ \beta_{LH} & \beta_{LL} \end{pmatrix},$$

where we expect $\beta_{HH}$ to be the largest value, $\beta_{LL}$ the lowest value, and $\beta_{LH}$ and $\beta_{HL}$ to be equal, thus defining a symmetrical matrix.

The mean-field equations for an SIS model considering the high–risk and low–risk groups would then be:

$$\begin{aligned} \frac{dI_H}{dt} &= \beta_{HH} S_H I_H + \beta_{HL} S_H I_L - \mu I_H \\ \frac{dI_L}{dt} &= \beta_{LH} S_L I_H + \beta_{LL} S_L I_L - \mu I_L, \end{aligned} \tag{1.30}$$

where the equations for the susceptible individuals can be omitted because $S_H = n_H - I_H$ and $S_L = n_L - I_L$. By assigning to each transmissibility rate a suitable value extracted from epidemiological data, this model would successfully represent an SIS dynamics with two additional compartments of the population according to their risk.

However useful, the latter approach is naive in the sense that it assumes that there are only two possible risk groups in the population. In the particular case where the risk has a direct translation to the amount of connections of an individual (as in the previous example of modeling HIV), the population can be subdivided in as many groups as number of different connectivity degrees in the population. Using the complex networks jargon, it translates to considering one risk compartment for each degree in the network. Opposite to the previous homogeneous models, where we assumed that each individual of the population behaved equally, now we are going to consider that two individuals are indistinguishable from each other if they have the same degree. Next, I will present the SIS formulation for this case, usually referred to as *degree-based heterogeneous mean-field* approach.

#### 1.4.2.1 *Degree-based heterogeneous mean-field approach*

Degree-based mean-field approach was the first theoretical approach for the analysis of general dynamical processes on complex networks. This approach assumes that all nodes of degree $k$ are statistically equivalent. This assumption implies that, instead of quantifying how many individuals belong to the compartment $\alpha$, we wish to know the density of individuals *with degree* $k$ in every compartment, $\rho_k^\alpha(t)$, or in other words, the probability that an individual with degree $k$ is in the compartment $\alpha$ at time $t$. Furthermore, the assumption also implies that any given vertex of degree $k$ is connected with the same probability $P(k'|k)$ to any node of degree $k'$. The variables $\rho_k^\alpha(t)$ are not independent, but fulfill the normalization condition $\sum_\alpha \rho_k^\alpha(t) = 1$. The total fraction of individuals in a

compartment $\alpha$ is $\rho^{\alpha}(t) = \sum_k P(k)\rho_k^{\alpha}(t)$. It is worth reminding that, by using this approach, we are not taking into consideration the full connectivity of the network, expressed by means of the adjacency matrix $a_{ij}$. Instead, only the degree and the two-vertex correlations of each node are preserved.

The SIS model can be approached using the previous approximation, as done in [169]. This model is described by means of the probability $\rho_k^I(t)$ that a node of degree $k$ is infected at time $t$. The SIS dynamical equation for $\rho_k^I(t)$ is then:

$$\frac{d\rho_k^I(t)}{dt} = -\rho_k^I(t)\mu + \beta k[1-\rho_k^I(t)]\sum_k P(k'|k)\rho_{k'}^I(t), \tag{1.31}$$

where $\beta$ and $\mu$ are the infectivity and recovery rates, respectively. The first term accounts for the recovery of nodes of degree $k$, proportional to the probability $\rho_k^I(t)$ that a node of degree $k$ is infected. The second term accounts for nodes entering the infectious state, and is proportional to the probability that a node of degree $k$ is susceptible $(1-\rho_k^I(t))$, times the probability that this node is connected to a node of degree $k'$ $(P(k'|k))$, multiplied by the probability that this last node is infected $(\rho_{k'}^I(t))$ and the infectivity parameter $\beta$. As usual, the effective spreading rate (or basic reproductive ratio $R_0$) is $\lambda = \beta/\mu$ and the density of susceptible individuals is calculated as $1-\rho_k^I(t)$.

The system of Eqs. 1.31 cannot be solved in a closed form for general degree correlations. However, by means of a linear stability analysis [28] we can obtain the value of the epidemic threshold $\lambda_c$:

$$\lambda_c = \frac{1}{\Lambda_M}, \tag{1.32}$$

where $\Lambda_M$ is the largest eigenvalue of the connectivity matrix, whose elements are:

$$C_{kk'} = kP(k'|k). \tag{1.33}$$

For any $\lambda = \beta/\mu > \lambda_c$ the epidemic enters in the endemic state. In the case of uncorrelated networks, we can say that $P(k'|k) = \frac{k'P(k')}{\langle k\rangle}$, and it is possible to obtain an explicit solution of the equations, see [169]. The epidemic threshold is then:

$$\lambda_c^{\mathrm{unc}} = \frac{\langle k\rangle}{\langle k^2\rangle}. \tag{1.34}$$

One of the conclusions extracted from this expression points out the crucial effect of degree heterogeneities in epidemic spreading. In networks with a power-law degree distribution with exponent $2 < \gamma \leq 3$, for which $\langle k \rangle \to \infty$ in the limit of a network of infinite size, the epidemic threshold tends asymptotically to zero. In other words, diseases spreading on these networks (a category in which most real networks fall in), are always in the endemic state.

For the case of SIR dynamics, the number of equations in the system increases, now accounting for the recovered state R. The partial densities of susceptible, infectious and recovered individuals are represented as $\rho_k^S(t)$, $\rho_k^I(t)$ and $\rho_k^R(t)$ respectively. The equations are the following (the ratio of susceptible is omitted due to the normalization condition $\rho_k^S(t) + \rho_k^I(t) + \rho_k^R(t) = 1$):

$$\frac{\rho_k^I(t)}{dt} = -\mu\rho_k^I(t) + \beta k \rho_k^S(t) \sum_{k'} P(k'|k)\rho_{k'}^I(t) \tag{1.35}$$

$$\frac{\rho_k^R(t)}{dt} = \mu\rho_k^I(t) \tag{1.36}$$

The value of the epidemic threshold is obtained equally to the SIS case, by performing a linear stability analysis. The same result is obtained, for the correlated case it is as in Eq. 1.32. For the uncorrelated case, in the case of annealed networks the same result as in Eq. 1.34 is obtained, and for static networks, one obtains an epidemic threshold of:

$$\lambda_c = \frac{\langle k \rangle}{\langle k^2 \rangle - \langle k \rangle}. \tag{1.37}$$

### 1.4.3 THE MICROSCOPIC MARKOV CHAIN APPROACH

The two previous deterministic approaches introduced, the homogeneous and heterogeneous mean-field formulations for the SIS and SIR, are approximations of the real spreading of an infectious disease which happens on a network of contacts[8]. Indeed, in the homogeneous mean-field (MF) formulation, local homogeneities of the ensemble are used to average the system, reducing drastically the

8 Even airborne diseases, that to do not need direct contact between individuals to transmit, can be understood as spreading on top of a network, if we define a contact between two people as a spatial proximity below a certain threshold.

degrees of freedom. In its heterogeneous counterpart (HMF), instead, the pool of individuals in the system is coarse-grained into degree classes and considers that all nodes in a degree class have the same dynamical properties. However useful, these two approaches are not designed to give information about the probability of individual nodes. Then, questions concerning the probability that a given node might be infected are not well posed in this framework. Instead, in order to obtain more information at the individual level of description, scientists have usually relied on Monte Carlo (MC) simulations, which have also been used to validate the results obtained using MF methods.

The approach that I will introduce next, called the *Microscopic Markov Chain Approach* (MMCA), introduced in [89] by Gómez et al., is a theoretical framework for the spreading of diseases in complex networks, focused on the individual properties of each node. This means that its aim is to calculate the probability that each node of the network is in each of the $\alpha$ compartments at each time step. Opposite to MF and HMF models, this approach *does* consider the underlying network of contacts, represented by the adjacency matrix $\mathbf{A}$, and the only assumption is that the probability that two nodes are infected is independent from each other (therefore ignoring second and higher order degree correlations).

To account for how the spreading is produced between neighbors, there are two main strategies: either by means of a *contact process* or a *reactive process*. The first considers that the contagion is expanded at a certain rate from an infected vertex to *one* neighbor at a time, while the second assumes that each vertex contacts *all* of his neighbors to try to infect them (i.e. a broadcast). The MMCA formulation explained next allows to range from a contact process to a reactive process, by quantifying the number of neighbors contacted at each time step with the ratio $\lambda$. This formulation is based on probabilistic discrete-time Markov chains and can be applied to weighted and unweighted complex networks. By using this approach, in addition to capturing the global dynamics of the different contact models and its associated critical behavior, it is possible to quantify the microscopic dynamics at the individual level by computing the probability that any node is infected in the asymptotic regime. Monte Carlo simulations agree with this formalism, and corroborate that it is able to correctly reproduce the whole phase diagram, even beyond the epidemic threshold.

#### 1.4.3.1 *MMCA formulation for the SIS model*

We start out by considering a network with $N$ nodes, whose connections are represented by the entries $a_{ij}$ of an $N$-by-$N$ adjacency matrix $\mathbf{A}$, which defines the structure of the underlying connectivity graph. Each node represents an individual (or any entity, in general), and a link represents the connection along which the infection spreads. In the case of weighted networks, we consider the entries $w_{ij}$ which account for the weights of the links. For the dynamics, we consider an SIS contact-based process. At each time step, an infected node makes a number $\lambda$ of trials to transmit the disease to its neighbors with probability $\beta$ per unit time. This forms a Markov chain where the probability of a node being infected depends only on the last time step. After some transient time, the previous dynamics sets the system into a stationary state in which the average density of infected individuals, $\rho$, defines the prevalence of the disease.

To find out the probability that any given node $i$ is infected at the stationary state, we denote by $r_{ij}$ the probability that a node $i$ is in contact with a node $j$, thus defining a new matrix $\mathbf{R}$. These entries represent the probabilities that existing links in the network are actually used to transmit the infection. The entries of this matrix are defined as:

$$r_{ij} = 1 - \left(1 - \frac{w_{ij}}{w_i}\right)^{\lambda_i}, \tag{1.38}$$

where $w_i = \sum_j w_{ij}$ is the total strength of node $i$. Of course, if nodes $i$ and $j$ are not neighbors, then $r_{ij}$ will be 0. At this point, we can decide to model the SIS as a contact process or a reactive process by tuning the value of $\lambda_i$ from $\lambda_i = 1$ (where the contact process is recovered) to $\lambda_i \to \infty, \forall i$, recovering the reactive process and effectively considering $r_{ij} = a_{ij}$, regardless of whether the network is weighted or not. Other prescriptions for $\lambda_i$ conform the spectrum of models that can be obtained using this unified framework.

Once the entries of the matrix $\mathbf{R}$ are known, we can compute the equations for the SIS, as follows:

$$p_i^I(t+1) = (1 - q_i(t))(1 - p_i^I(t)) + (1 - \mu)p_i^I(t), \tag{1.39}$$

where $\mu$ is, as usual, the spontaneous recovery rate, and term $q_i(t)$ is the probability of node $i$ not being infected by any neighbor:

$$q_i(t) = \prod_{j=1}^{N} (1 - \beta r_{ij} p_j(t)). \tag{1.40}$$

The first term on the right-hand side of the equation is the probability that node $i$ is susceptible $(1 - p_i^I(t))$ and is infected by at least a neighbor $(1 - q_i(t))$, while the second term stands for the probability that node $i$ is infected at time $t$ and does not recover. As usual, $p_i^S(t) = 1 - p_i^I(t)$. The phase diagram of the model is simply obtained by solving the system formed by Eq. 1.39 for $i = 1, \ldots, N$ at the stationary state:

$$p_i^I = (1 - q_i)(1 - p_i^I) + (1 - \mu)p_i^I, \tag{1.41}$$

which has always the trivial solution $p_i = 0$, $\forall i = 1, \ldots, N$. Other non-trivial solutions are reflected as non zero fixed points of Eq. 1.41 and can be easily computed numerically by iteration. The macroscopic order parameter in which we are interested is given by the expected infection density $\rho = \frac{1}{N} \sum_i^N p_i$.

To illustrate the results that can be obtained by using this method, the authors of [89], performed a Monte-Carlo simulation on top of a certain topology, and compare it with the numerical results obtained with the MMCA formulation. This result can be seen in Fig. 1.16, where the initial fraction of infected nodes is $\rho_0 = 0.05$. At each time step an infected node $i$ infects with the same probability $\beta$ all its neighbors and recovers at a rate $\mu$. The simulation runs until a stationary state for the density of susceptible individuals, $\rho(t)$ is reached. As we can see, the agreement between the MMCA numerical result and the MC simulation is flawless. Moreover, the formalism also captures the microscopic dynamics as given by the $p_i$'s, see Fig. 1.16 (inset).

Once the equations for the SIS dynamics are defined, we can also calculate the expression for the epidemic threshold. Let us start by assuming the existence of a critical point $\beta_c$ for fixed values of $\mu$ and $\lambda_i$ such that $\rho = 0$ if $\beta < \beta_c$ and $\rho > 0$ when $\beta > \beta_c$. The calculation of this critical point is performed by considering that when $\beta \to \beta_c$, the probabilities $p_i \approx \epsilon_i$, where $0 \leq \epsilon_i \ll 1$, and then after substitution in Eq. 1.40 one gets:

$$q_i \approx 1 - \beta \sum_{j=1}^{N} r_{ji} \epsilon_j. \tag{1.42}$$

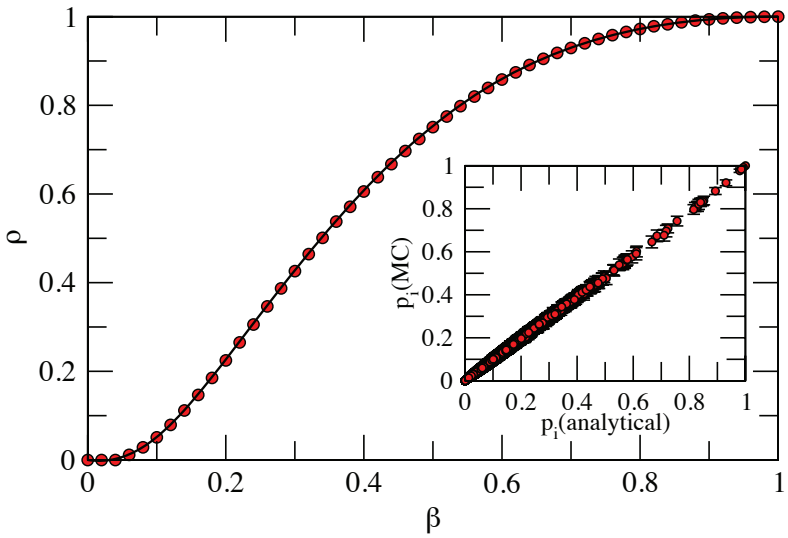



Figure 1.16: Average fraction of infected individuals, $\rho$, as a function of the spreading rate $\beta$ for a network of $N = 10^4$ nodes. The symbols correspond to MC simulations of the SIS model on top of a random scale-free network with $\gamma = 2.7$ (error bars are smaller than the size of the symbol) and the lines stand for the analytical solutions of the MMCA (with $\lambda \to \infty$, thus recovering a reactive process). **Inset.** Scatter plot of the probability that a node $i$ is infected, using the results of the MC simulations (y-axis) and the MMCA numerical solutions (x-axis). Both plots have $\mu = 1$ and $\beta = 0.1$. *Original source: [89]. Reprinted with permission.*

Inserting Eq. 1.42 in Eq. 1.41, and neglecting second order terms in $\epsilon$ we get

$$\sum_{j=1}^{N} \left( r_{ji} - \frac{\mu}{\beta} \delta_{ji} \right) \epsilon_j = 0 \qquad \forall i = 1, \ldots, N \tag{1.43}$$

where $\delta_{ij}$ stands for the Kronecker delta. The system 1.43 has non trivial solutions if and only if $\mu/\beta$ is an eigenvalue of the matrix $\mathbf{R}$. Since we are looking for the onset of the epidemic, the lowest value of $\beta$ satisfying 1.43 is

$$\beta_c = \frac{\mu}{\Lambda_{\max}} \tag{1.44}$$

where $\Lambda_{\max}$ is the largest eigenvalue of the matrix $\mathbf{R}$ and $\beta_c$ defines the epidemic threshold of the disease spreading process.

As we can see, this methodology is powerful, effective and conceptually simple, and the only assumption taken is that the probabilities of being infected $p_i^I$ are independent random variables. Luckily, this hypothesis turns out to be valid in the vast majority of complex networks because the inherent topological disorder makes dynamical correlations not persistent. The formulation is easily extended to other kinds of models other than a simple SIS, as we will see in Chapter 3, where we will use this formulation to solve an epidemic spreading problem on top of multiplex networks.

UNIVERSITAT ROVIRA I VIRGILI
FROM COMMUNITY STRUCTURE TO THE PHYSICS OF MULTIPLEX NETWORKS
Clara Granell Martorell

# 2

# ON THE ANALYSIS OF THE MESOSCOPIC STRUCTURE OF NETWORKS

Most real world systems display a pattern of connectivity that presents community structure, meaning that entities are organized in groups of highly dense connectivity. Unveiling such structure is key to the understanding why the system is organized as it is and it supposes a way of learning about the implications of the structure in the network dynamics. However, as previously pointed out, the problem of finding communities is ill-posed due to the variety of possible definitions, and furthermore it is known to be a computationally costly problem to solve. Designing new methods for detecting the community structure of networks is the goal of many scientists working in our field, and although there are some well-known established techniques to address this question, it is still an open problem.

One of the open issues is the design of algorithms suited to analyze any kind of networks. Networks which account only for the presence or absence of connections between nodes (unweighted networks) are certainly useful, but most of the networks of our interest display a more complicated array of features. Some systems need to be represented using directed edges (e.g. food webs), signed edges (e.g. correlation networks), or even more complicated representations such as bipartite or time varying networks. Using the appropriate community detection technique for each type of network is crucial if we wish to respect the nature of our data, and neglecting it would lead to a misleading interpretation of our system. Designing versatile algorithms is then a matter of great importance. Equally important is to be able to put the algorithms to test, as most of the time we do not have any previous information on the true structure of communities of our data. This is the purpose of benchmarks – toy models of networks with a planted community structure that our algorithm will aim to recover. Indeed, understanding the behavior and limitations of each algorithm is needed if we intend to give meaning to

the substructure that we have found. As mentioned, each community detection algorithm comes with an implicit definition of community, that may or may not coincide with the type of communities we are looking for. Identifying the weak points and non-intuitive behaviors of some techniques (e.g. the resolution limit of modularity) allows us to create more refined methods that can give us a better insight on the structure of real networks.

## 2.1 EXTENSION OF THE AFG ALGORITHM TO THE CASE OF SIGNED NETWORKS AND CALCULATION OF ITS BOUNDARIES

This section is devoted to present an extended formulation for the AFG multi-resolution algorithm (see Sec. 1.2.3.2 for its introduction), which will allow computing modularity in multiple resolutions even in the case where networks have signed weights. As stated previously, the AFG multi-resolution algorithm is a technique devoted to overcome the resolution limit, by allowing access to the different topological scales in a network. However, the algorithm presents some limitations, as it is only able to operate on top of unweighted and weighted networks. This, however, is insufficient to analyze networked data which contains signs. There are multiple occasions in which we will be interested in taking into account signed links. For instance, networks resulting from the calculation of the correlation between some features of the nodes will contain positive and negative valued edges. Similarly, social networks may have negative interactions accounting for conflict or opposition between people. Here we follow the intuition that a negative link between two nodes should be treated by the algorithm as a repulsive force that contributes in placing such nodes in different communities.

The AFG algorithm is based on the modularity formulation, thus the extension of modularity to weighted signed networks is our starting point.

### 2.1.1 GENERALIZATION OF MODULARITY TO WEIGHTED SIGNED NETWORKS

The generalization of modularity to any network, with weighted, directed and signed values of the weights [93] is as follows. Let us suppose that we have a

weighted undirected complex network with weights $w_{ij}$ as above. The relative strength $p_i$ of a node

$$p_i = \frac{w_i}{2w}\,, \tag{2.1}$$

may be interpreted as the probability that this node makes links to other ones, if the network were random. This is precisely the approach taken by Newman and Girvan to define the modularity null case term, which reads

$$p_i p_j = \frac{w_i w_j}{(2w)^2}\,. \tag{2.2}$$

The introduction of negative weights destroys the probabilistic interpretation of $p_i$, since in this case the values of $p_i$ are not guaranteed to be between zero and one. The problem is the implicit hypothesis that there is only one unique probability to link nodes, which involves both positive and negative weights. To solve this problem, we have to introduce two different probabilities to form links, one for positive and the other for negative links.

Let us formalize this approach. First, we separate the positive and negative weights:

$$w_{ij} = w_{ij}^{+} - w_{ij}^{-}\,, \tag{2.3}$$

where

$$w_{ij}^{+} = \max\{0, w_{ij}\}\,, \tag{2.4}$$

$$w_{ij}^{-} = \max\{0, -w_{ij}\}\,, \tag{2.5}$$

we use these expressions because in principle we do not know if $w_{ij}$ is positive or negative. The positive and negative strengths are given by

$$w_i^{+} = \sum_j w_{ij}^{+}\,, \tag{2.6}$$

$$w_i^{-} = \sum_j w_{ij}^{-}\,, \tag{2.7}$$

and the positive and negative total strengths by

$$2w^+ = \sum_i w_i^+ = \sum_i \sum_j w_{ij}^+ , \tag{2.8}$$

$$2w^- = \sum_i w_i^- = \sum_i \sum_j w_{ij}^- . \tag{2.9}$$

Consequently,

$$w_i = w_i^+ - w_i^- \tag{2.10}$$

and

$$2w = 2w^+ - 2w^- . \tag{2.11}$$

With these definitions at hand, the connection probabilities with positive and negative weights are respectively

$$p_i^+ = \frac{w_i^+}{2w^+} , \tag{2.12}$$

$$p_i^- = \frac{w_i^-}{2w^-} . \tag{2.13}$$

Now, there are two terms which contribute to modularity: the first one takes into account the deviation of actual positive weights against a null case random network given by probabilities $p_i^+$, and the other is its counterpart for negative weights. Thus, it is useful to define

$$Q^+ = \frac{1}{2w^+} \sum_i \sum_j \left( w_{ij}^+ - \frac{w_i^+ w_j^+}{2w^+} \right) \delta(C_i, C_j) , \tag{2.14}$$

$$Q^- = \frac{1}{2w^-} \sum_i \sum_j \left( w_{ij}^- - \frac{w_i^- w_j^-}{2w^-} \right) \delta(C_i, C_j) . \tag{2.15}$$

The total modularity must be a trade off between the tendency of positive weights to form communities and that of negative weights to destroy them. If we want that $Q^+$ and $Q^-$ contribute to modularity proportionally to their respective positive and negative strengths, the final expression for modularity $Q$ is

$$Q = \frac{2w^+}{2w^+ + 2w^-} Q^+ - \frac{2w^-}{2w^+ + 2w^-} Q^- . \tag{2.16}$$

Which, after substituting $Q^+$ and $Q^-$, it reads:

$$Q = \frac{1}{2w^+ + 2w^-} \sum_i \sum_j \left[ w_{ij} - \left( \frac{w_i^+ w_j^+}{2w^+} - \frac{w_i^- w_j^-}{2w^-} \right) \right] \delta(C_i, C_j) \,. \tag{2.17}$$

The main properties of Eq. 2.17 are the following: without negative weights, the standard modularity is recovered; modularity is zero when all nodes are together in one community; and it is antisymmetric in the weights, i.e. $Q(C, \{w_{ij}\}) = -Q(C, \{-w_{ij}\})$ .

The extension to directed networks [9] is simply obtained by the substitutions in Eq. 2.17 of

$$w_i^\pm \quad \to \quad w_i^{\pm,\text{out}} = \sum_k w_{ik} \,, \tag{2.18}$$

$$w_j^\pm \quad \to \quad w_j^{\pm,\text{in}} = \sum_k w_{kj} \,. \tag{2.19}$$

### 2.1.2 MESOSCALES ANALYSIS FOR WEIGHTED SIGNED NETWORKS

The extension of the multiple resolution method to the general case of weighted signed networks follows the same idea as the generalization of modularity. The method relies on the introduction of a magnitude $r$ that we call *resistance*, represented by a self-link for each node, that stands for the opposition of a node to belong to a group, in terms of modularity contributions. We tune the resistance uniformly for all nodes because in this way the functional form of the strength distribution is preserved and does not distort the relative structural properties of nodes. More precisely, the prescription of signed modularity $Q_r$ at different resolution scales tagged by $r$ consists in substituting in the following expressions in Eq. 2.17

$$w_{ij} \quad \to \quad w_{ij} + r\delta_{ij} \,, \tag{2.20}$$

$$w_i^\pm \quad \to \quad w_i^\pm + r^\pm \,, \tag{2.21}$$

$$2w^\pm \quad \to \quad 2w^\pm + Nr^\pm \,, \tag{2.22}$$

where

$$r = r^+ - r^- \,, \tag{2.23}$$

and

$$r^{+} = \max\{0, r\}, \tag{2.24}$$
$$r^{-} = \max\{0, -r\}. \tag{2.25}$$

Thus, adding the positive and negative contributions of the resistance parameter to Eq. 2.17, it reads:

$$Q = \frac{1}{2w^{+} + 2w^{-} + N|r|} \times \sum_{ij} B_{ij}\delta(C_i, C_j), \tag{2.26}$$

where

$$B_{ij} = \left[ w_{ij} + r\delta_{ij} - \left( \frac{(w_i^{+} + r^{+})(w_j^{+} + r^{+})}{2w^{+} + Nr^{+}} - \frac{(w_i^{-} + r^{-})(w_j^{-} + r^{-})}{2w^{-} + Nr^{-}} \right) \right]. \tag{2.27}$$

### 2.1.3 CALCULATION OF THE BOUNDARIES OF THE MESOSCALE

The method to unveil the whole mesoscale of a complex network consists in the optimization of $Q_r$ for different values of the resistance parameter $r$. The screening of this parameter will eventually reveal different optimal partitions (found by heuristic algorithms to detect community structure) that represent intermediate topological scales of the complex network. An important issue is the calculation of the value of the resistance parameter at the boundaries of the mesoscale. Indeed, an accurate calculation of such limits is necessary to design an algorithm that is able to screen the whole mesoscale in an efficient way. The boundaries of the mesoscale are the *macroscale* —a partition in which all nodes belong to the same community—, and the *microscale* — a partition in which each node is isolated in its own community. In practice, calculating the value of $r$ that recovers the two limiting partitions is equivalent to finding two values of the self-loops, $r_{\min}$ and $r_{\max}$, for which the $Q_{\mathrm{AFG}}(r)$ modularity is maximum at the macroscale and microscale respectively.

Next I present the mathematical proofs of the physical limiting cases of the resistance for weighted signed networks. To determine $r_{\max}$ we look for a value of the resistance such that the increment in modularity when joining any pair of

vertices in the same community is negative, and the contrary for $r_{\min}$. The idea is the following: if $r > 0$ and all the non-diagonal terms ($i \neq j$) of Eq. 2.17 are negative,

$$w_{ij} \leq \frac{(w_i^+ + r)(w_j^+ + r)}{2w^+ + Nr} - \frac{w_i^- w_j^-}{2w^-}\,, \quad \forall i \neq j\,, \tag{2.28}$$

then the maximum of $Q_r$ is achieved with the partition which satisfies $\delta(C_i, C_j) = 0$ for all $i \neq j$, i.e. the partition in which all nodes are isolated. Eqs. 2.28 form a system of second order inequations in $r$. After some algebra, it can be shown that $r_{\max}$ is the lowest value of $r$ for which the following set of inequalities per link (denoted $ij$) is satisfied:

$$\min_{r,ij}[Ar^2 + B_{ij}r + C_{ij} \leq 0] \tag{2.29}$$

where

$$A = -2w^- \tag{2.30}$$
$$B_{ij} = N(2w^- w_{ij} + w_i^- w_j^-) - 2w^-(w_i^+ + w_j^+) \tag{2.31}$$
$$C_{ij} = 2w^- 2w^+ w_{ij} + 2w^+ w_i^- w_j^- - 2w^- w_i^+ w_j^+ \tag{2.32}$$

Equivalently, if $r < 0$ and all the non-diagonal terms ($i \neq j$) of Eq. 2.17 are positive,

$$w_{ij} \geq \frac{w_i^+ w_j^+}{2w^+} - \frac{(w_i^- - r)(w_j^- - r)}{2w^- - Nr}\,, \quad \forall i \neq j\,, \tag{2.33}$$

the maximum of $Q_r$ is achieved with the partition which satisfies $\delta(C_i, C_j) = 1$ for all $i \neq j$, i.e. the partition in which all nodes are together in the same community. Thus, to determine a lower bound of $r_{\min}$ we look for the largest value of $r$ satisfying

$$\max_{r,ij}[Ar^2 + B_{ij}r + C_{ij} \geq 0] \tag{2.34}$$

where

$$A = 2w^+ \tag{2.35}$$
$$B_{ij} = N(2w^+ w_{ij} - w_i^+ w_j^+) + 2w^+(w_i^- + w_j^-) \tag{2.36}$$
$$C_{ij} = 2w^+ 2w^- w_{ij} - 2w^- w_i^+ w_j^+ + 2w^+ w_i^- w_j^- \tag{2.37}$$

The value of $r$ obtained from Eqs. 2.41 is only a lower bound of the exact $r_{\min}$, since these equations are only sufficient conditions for the existence of a unique community holding all the nodes of the network (not all terms in Eq. 2.17 need to be positive in the $r_{\min}$ limit). On the other hand, Eqs. 2.29 are necessary and sufficient conditions, and thus the $r_{\max}$ found is the exact value.

## 2.2 THE COMPLEX NETWORKS APPROACH TO DATA CLUSTERING

The complex networks toolset has been proven to be very versatile and useful in a variety of problems, some of them outside the complex system domain. After extending the AFG multi-resolution community detection algorithm to the case of signed networks, we are now able to show one of the possible applications of such method in a traditional problem in computer science, which is the unsupervised classification of patterns into groups, or *data clustering*. We will show that the formulation for signed networks allows us to approach solving this problem without neglecting the signs in the calculations of the similarities between elements. We will apply our method to the Iris dataset, a classical benchmark in this topic, and we will show that the results obtained using a complex networks approach are competitive with the best data clustering techniques available.

### 2.2.1 UNSUPERVISED DATA CLUSTERING

The problem of unsupervised data clustering consists in classifying elements so that two data points belonging to the same cluster are more similar between them than with elements in a different cluster. An element, or pattern, is a vector of features (usually understood as a point in a multidimensional space) that describes the item we wish to classify. The goal of the process of data clustering is to organize these patterns finding a partition of the sample according to the natural classes that are present in it. Data clustering has been the subject of interest in many disciplines where the mining of raw information is crucial to understand some phenomenon or gain insight into a system. It has applications in several fields such as pattern recognition, astronomic classification, biological taxonomy, marketing, and many more [82].

The methodology used to obtain the clusters from the raw data is as follows. First of all, a representation of the patterns has to be chosen, and also a feature selection or extraction is performed. Feature selection means choosing, from all the available features, those that will make easier the process of clustering, leaving the redundant, correlated and less informative features out of the analysis. On the other hand, feature extraction consists in transforming the original data set to a new one containing only the most relevant information. This first step has to be done carefully, as the result of the clustering depends directly on the quality of this procedure. Secondly, the similarity or dissimilarity between each pair of patterns has to be computed, which is often done by using a measure of distance. The result of this step is the similarity matrix, which using the mapping to complex networks can be understood as a graph, where each node is a pattern and the links are the similarities between them [112]. Finally, it is time for the main step of the process, the grouping (or clustering) algorithm, which will decompose the similarity matrix and return the groups of data.

The problem of clustering is inherently ill-posed, i.e. any data set can be clustered in drastically different ways, with no clear criterion for preferring one clustering over another. In particular, in the case of unsupervised approaches, a satisfactory clustering of data depends on the desired resolution which determines the number of clusters and their size. For example, $k$-means clustering fixes a priori the number of groups ($k$), which implies indeed a certain resolution of the clustering. Other algorithms such as hierarchical clustering [118] group the patterns extending the measure of distance between them to distances between clusters of patterns. This process generates a complete dendrogram. Cutting the dendrogram at different heights we obtain different partitions of the data, all them hierarchically nested. In this situation the following question arises: To what resolution should one look at the data to find a scientific meaning in the classification? We claim that the answer to this question is totally dependent on the final purpose of the classification process, and that the concept of best solution should be reconsidered. Different partitions will be representative of properties of the data at different scales and then all of them are worth to be studied. As we will see, by using a multi-resolution community detection algorithm to approach this problem instead of a single resolution one, we will have access to all the scales of organization of our data. Furthermore, by plotting all partitions against the resistance parameter we will obtain an assessment on which scale is the most meaningful for our analysis.

### 2.2.2 ON THE PREPROCESSING OF THE DATA

Here we briefly review the two stages of the data clustering preprocess before performing the clustering, for an extended revision see [102]. In brief, it consists in two stages concerning the data representation and the definition of similarity measures between data points.

The first stage of data clustering is to represent the data which we are interested in clustering. These data are usually obtained experimentally, and our first task is to prepare them properly to give the best possible result when applying the clustering algorithm. A good representation of the patterns will result in a simple and easy clustering process, while a poor representation can lead to complex groups whose structure is difficult or impossible to ascertain. It is worth then to invest some time analyzing the original data to see if one can make a proper pretreatment. Given that any clustering process will try to find regularities among the data, a good pretreatment should facilitate the process by filtering noisy or redundant information, and reducing the data dimensionality to simplify its computational handling. Usually data are represented as vectors of features, being those categorical or numerical. Without loss of generality, in what follows we will assume that the clustering is intended on vectors of numerical features.

One of the techniques to preprocess the data is feature selection. It will be necessary to apply a feature selection algorithm when some of the features are correlated with each other. In this case, these variables provide redundancy into the system and can introduce a bias towards the final classification based on differences in other not-correlated features. Another scenario where this is useful are cases in which we have an excessive number of variables and a discriminatory elimination could enhance their handling. Among the different methods for feature selection, we have for example, forward selection/backward elimination: In forward selection, we grow subsets of features depending on the classification obtained, while in backward elimination, we start with all the variables and we eliminate those less promising also according with the classification obtained. Another technique is the decision tree, where we consider the problem of variable selection as a decision problem. Once this analogy is assumed, the decision consists in finding out which subset of variables is more appropriate. As in any decision problem based on trees, the result of the selection will depend on the utility functions used. A third alternative is the naive Bayes classifier, which is a simple probabilistic classifier based on the application of Bayes' theorem.

In the context of variable selection this method can involve certain assumptions about dependence or independence of variables and compute their conditional probabilities. Finally, it is worth mentioning the neural networks approaches, e.g. self-organized Kohonen maps, in which the c-plane map of variables is analyzed in order to determine which of those variables can offer better differentiation groups.

There may be some cases in which all features are significant or we have a small amount of features and the elimination of any of them would cause a great loss of information. In these situations, a feature extraction method is more adequate than a feature selection technique. A feature extraction method is an algorithm that takes as input the original features and mixes and/or merges them producing a set of new categories that can be filtered and analyzed in the same way as the original data. Examples of feature extraction algorithms are: Principal Component Analysis (PCA) [114], a method aimed to perform a linear transformation of the data converting a set of correlated variables into a new set of less correlated variables called principal components. The first principal component recovers the maximum variance, the second component retrieves the second highest variance and so on, until all have described the variability of the original data. Algebraically, the process involves finding a basis of orthogonal vectors (the principal components) in the $n$-dimensional space of the original variables, such that the length of the components provides information on the volume and distribution of the data in different directions of the space. In this way, using the main components of the data instead of all the original features, we can capture most of the information in a reduced set of variables. This is one of the most widely used techniques for the purpose of feature extraction. Other alternatives apart from PCA include nonlinear projections such as self-supervised backpropagation in neural networks or Independent Component Analysis (ICA).

The second stage of the process of clustering is to calculate the similarity (or dissimilarity) between patterns according to a similarity measure, which is usually based on a distance function. The representation of these similarities forms a square matrix of size $N \times N$, where $N$ is the total number of patterns of the dataset. The similarity matrix can be understood as a complete weighted graph where each node is regarded as one of the patterns and the weight of the link between them informs about their similarity. Note that if the similarity measure used is not symmetric, then the graph should be directed. Once the similarity matrix is obtained, it is time to apply the grouping algorithm that

ends the data clustering process. Also, one can apply graph based community detection algorithms to perform this step, as we will see next.

### 2.2.3 COMPLEX NETWORKS COMMUNITY DETECTION APPROACH TO DATA CLUSTERING

To show the ability of multi-resolution community detection methods to solve the problem of unsupervised data clustering, we have chosen to study the classical benchmark of the Iris data set. This data set was presented by Sir Ronald Aylmer Fisher in 1936 [73], and consists of 150 patterns corresponding to three different classes of Iris flowers: Setosa, Versicolor and Virginica. Four features, the width and length of petal and sepal, form each pattern. Plots for the cross-variables and subspecies are represented in Fig. 2.1. The unsupervised classification of this dataset still remains a major challenge in artificial intelligence and statistical theory, because of the patterns' organization: while one of the classes is linearly separable and thus easily to classify by any elemental classification algorithm, the other two classes are not linearly separable and consequently hamper the whole classification problem.

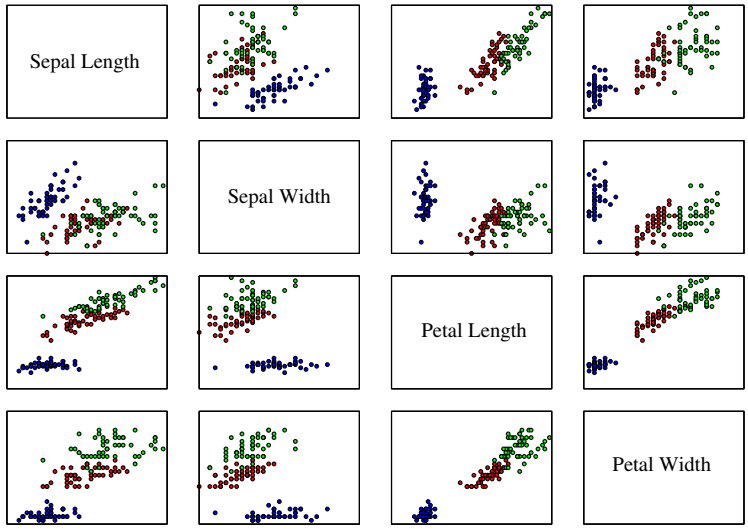



Figure 2.1: Feature vectors of the Iris data set. Colors correspondence are: setosa-blue, versicolor-red, and virginica-green.

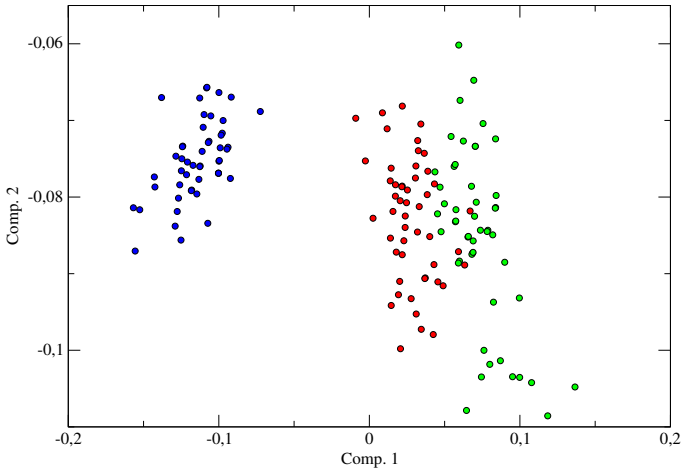



Figure 2.2: Two principal components of the PCA analysis on the Iris dataset. Colors correspondence are: setosa-blue, versicolor-red, and virginica-green. The separation of the pattern classes seems more clear in this projection.

Following the steps of data clustering explained above, we first performed a feature extraction/selection process. The idea here is simply to follow the workflow in any clustering problem, where the high dimensionality of the data and its redundancy is a main concern. In the particular case we analyze, we can use all the original data with no computational stress, however we propose to address the feature extraction using PCA which is the most common approach in many scenarios. We performed the principal component analysis of the four features that form each pattern, and choose to work with the two principal components corresponding to the largest part of the data variance. In Fig. 2.2 a representation of these two components is shown. Based on these two variables, we propose to build up a similarity matrix of the euclidean distances between patterns components with respect to the center of mass of the data set in this space. For any pair of flower samples $i$ and $j$, we define the similarity $s_{ij} = \bar{d} - \|x^i - x^j\|)$, where $\bar{d}$ stands for the average distance of the set, and $\|\cdot\|$ is the euclidean distance between the feature vectors of each flower. The resulting similarity matrix is interpreted as a signed weighted network whose communities will, in principle, reproduce the right clustering of the data.

The result of the multiple resolution algorithm on the two main components of the Iris dataset is shown in Fig. 2.3. As suggested in the prescription of the AFG algorithm, we look at the longest (thus more stable) plateaus of this plot in

seek of the partition that we will take as our result. As we can see, the longest plateau in terms of the resistance interval values is formed by those partitions that divide the dataset into two communities. This is not a surprising fact, as we know beforehand that one of the three classes of flowers is linearly separable, and then this partition makes good sense, since there is one for the Setosa class and the other one containing the Versicolor and Virginica. However, the second longest plateau is the one formed by the three community partitions, and if we analyze the most resistant of them, we realize that it largely corresponds to the biological taxonomy of the flowers. To be specific, if we calculate the success as the number of correctly classified nodes divided by the total number of nodes, we achieve for the most resistant partition of three communities a 94,67% of success compared to the correct biological taxonomy.

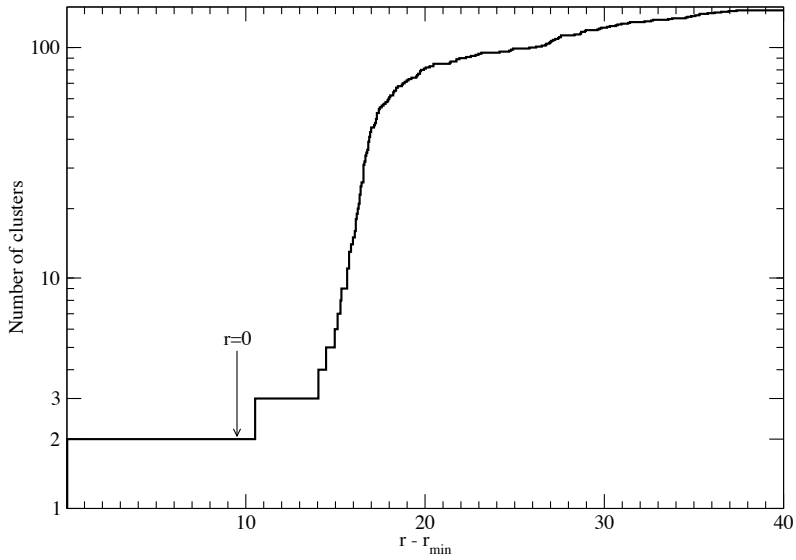



Figure 2.3: Result of the application of the AFG multi-resolution community detection on the Iris dataset. Vertical axis accounts for the number of groups in each partition and horizontal axis accounts for the resistance parameter. Intuitively, partitions unchanged for more values of the resistance parameter are more stable and thus a plausible solution for the division of the dataset into groups.

### 2.2.4 COMPARISON WITH REICHARDT & BORNHOLDT MULTI RESOLUTION ALGORITHM

To compare the results obtained with the AFG method, we perform the same community analysis using a widely used multi-resolution algorithm, the Reichardt & Bornholdt algorithm [180]. In this algorithm, a parameter $\gamma$ is introduced in front of the null-case term to tune its relative importance against the real network, with $\gamma$ ranging from $\gamma_{\min}$ to $\gamma_{\max}$. Optimizing modularity for each value of this parameter, we are able to plot the mesoscales obtained, see Fig. 2.4 for a portion of such mesoscales. We observe, as we did in the case of AFG, that the partition in two communities holds for a large range of values of the multi-resolution parameter; however, the variations of $\gamma$ do not ensure a monotonic behavior of the number of clusters as a function of $\gamma$. Without negative weights, the macroscale is recovered at $\gamma_{\min} = 0$, and the microscale at the $\gamma_{\max}$ which makes all modularity terms negative. The existence of $\gamma_{\max}$ is guaranteed by the fact that all null-case terms are positive. However, the addition of negative weights to the network makes it possible to have both positive and negative null-case terms, which does not ensure the recovery of macro and microscale. Therefore, RB signed modularity may not cover the whole mesoscale.

This is experimentally confirmed in Fig. 2.5 for the Iris data set, where a larger interval of the $\gamma$ parameter has been analyzed. While Fig. 2.4 only shows the useful part of the mesoscales range, where the number of clusters goes from 2 to 73 ($\gamma \in [0.0, 4.2]$), in Fig. 2.5 we show the inability of the RB algorithm to find the macroscale (microscale) for lower (larger) values of $\gamma$.

### 2.2.5 COMPARISON WITH A HIERARCHICAL CLUSTERING METHOD

For the sake of completeness, next we present a comparison of the results obtained with the AFG multi-resolution algorithm and a traditional approach known as hierarchical clustering (HC). This method, used in data mining and statistics, seeks to build a hierarchy of clusters according to some similarity measure between the elements. In particular, we used a hierarchical clustering agglomerative approach, constructed using complete linkage, which means that the distance between groups is defined as the distance between the most distant pair of individuals, one from each group. In other words, the distance between two clusters is given by the value of the longest link between the clusters. At each stage of

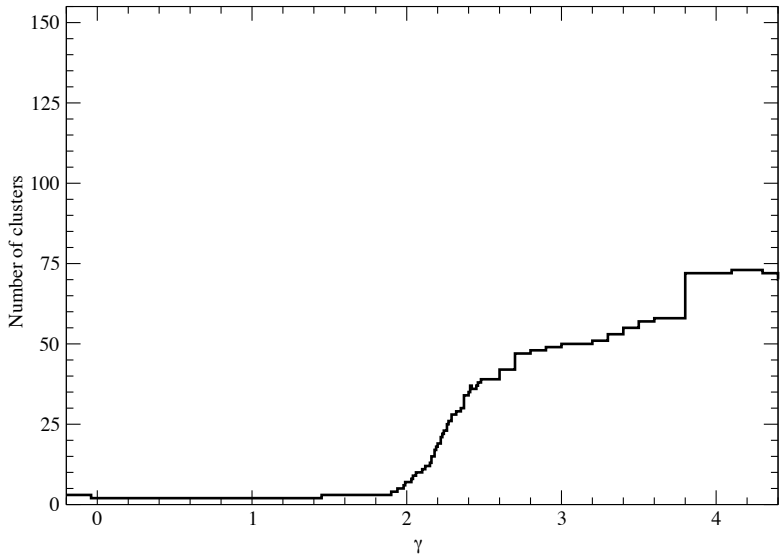



Figure 2.4: Mesoscales of the RB multi-resolution algorithm for the clustering of the Iris dataset.

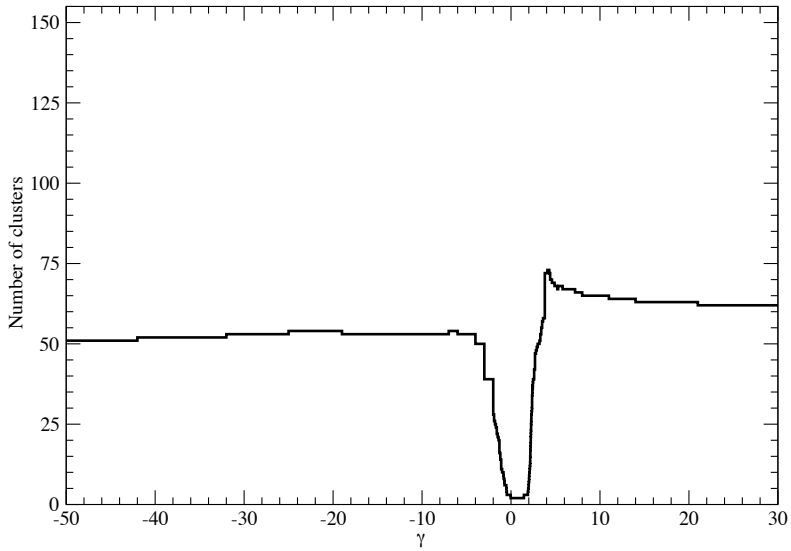



Figure 2.5: Expanded results of the RB algorithm for the clustering of the Iris dataset. For a larger range of $\gamma$ values we are able to observe the non-monotonic behavior as well as the impossibility to access the whole mesoscale.

hierarchical clustering, the clusters at minimum distance are merged. Moreover, instead of using the standard pair-group hierarchical clustering approach, we take advantage of a recent development [71] that allows to solve the non-uniqueness problem when there are tied distances during the agglomeration process. The result, known as a *multidendrogram*, is shown in Fig. 2.6. We plot the tag number of each sample of flower at the leaves of the tree. The interpretation of the multidendrogram is as follows: starting from the root of the tree, the branches split the data in clusters at different heights, which account for the ultra-metric distance between the divided clusters.

The comparison between the two methods can be done by computing the multiple scales of the topology in terms of community structure, by screening the distances in the dendrogram and comparing them to the values of $r$ in the AFG method (shown previously in Fig. 2.3). The mesoscale for the HC approach is shown in Fig. 2.7, where we plot the number of clusters in each partition as a function of the ultra-metric distances between the clusters. As we observe, the hierarchical clustering approach defines also two main resolution levels corresponding to two and three clusters partitions, respectively. In order to quantitatively compare the hierarchical clustering method to our multi-resolution approach, we define two measures. The first measure is the success ratio, which is the percentage of correctly classified nodes when comparing the partition obtained with the original classification made by biologists using more features of the flowers. In this case and for the partition in three clusters, both HC and AFG methods are tied to $94,67\%$ of success, which corresponds to a mismatch of eight flowers in total. The second measure we contemplate is the Jaccard index [110], which is defined as the fraction of pairs of patterns in the same cluster in one partition which are also in the same cluster in the other partition. The larger the fraction of same cluster co-occurrences, the better the quality of the agreement. For the case of the division in three groups, the Jaccard score obtained by the HC is 0.8180, while AFG obtained a score of 0.8194.

We can conclude that the application of a community detection algorithm to the problem of data clustering is able to compete with a well established hierarchical clustering technique. These results are encouraging, and point out that the mapping of clustering problems to networks' structural analysis is a field worth to be explored.

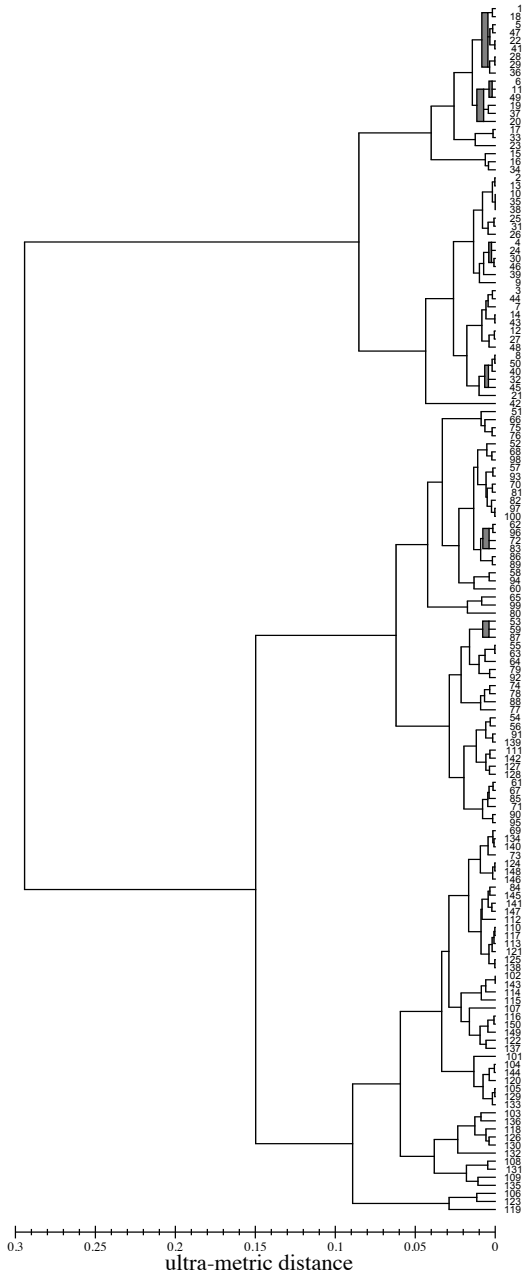



Figure 2.6: Dendrogram obtained by applying a hierarchical clustering algorithm to the Iris dataset. Horizontal axis is the ultra-metric distance.

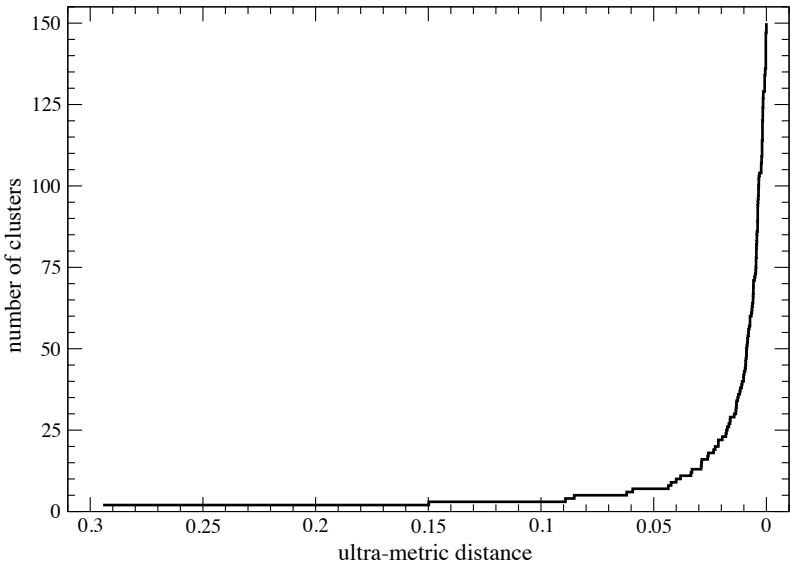



Figure 2.7: Hierarchical clustering mesoscales of the Iris dataset. Vertical axis is the number of clusters in each partition and horizontal axis is the ultra-metric distance, which also gives us a sense of stability of partitions.

## 2.3 HIERARCHICAL METHOD TO OVERCOME THE RESOLUTION LIMIT OF MODULARITY

Multi-resolution modularity optimization algorithms were designed to provide a tool able to unravel the whole mesoscopic structure of networks. The possibility of accessing the whole structure has proven to have multiple applications, being one of them the previously presented application to data clustering, among many others. Furthermore, this approach allowed avoiding the well-known resolution limit of modularity, a limit beyond which optimization of modularity is unable to identify certain modules as communities. This limit is due to the fact that modularity fixes a global scale that can be appropriate for some networks but not for others, being specially unsuited for networks whose communities are very small compared to the size of the whole network. Indeed, multi-resolution algorithms introduce a resolution parameter in the modularity formulation, allowing access to different scales and being able to dodge the resolution limit.

However, authors Lancichinetti & Fortunato pointed out in [126] that the problem of modularity finding counterintuitive partitions is extended beyond the problem of the limit of resolution. In this work, the authors show a new setup for which even multi-resolution algorithms are unable to identify the correct partitions. Networks that suffer from this problem, which I will refer to as the problem of *splitting and merging*, are those in which very different size scales for the communities coexist. In the following, I will present a setup that illustrates this problem, as well as the proposed new method designed to overcome it.

### 2.3.1 THE PROBLEM OF SPLITTING AND MERGING COMMUNITIES

To illustrate this problem, let me point the attention of the reader to Fig. 2.8 (henceforward referred to as the LF benchmark), a network formed by three easily recognizable subgroups. It consists in a dense Erdős-Renyi network of 400 nodes with $\langle k \rangle = 100$ and two cliques of 13 nodes each, where the three subgroups are connected with a single link.

The purpose of this benchmark is to show that even multi-resolution algorithms fail to correctly classify the nodes of the network in the three natural subgroups. The result obtained by optimizing modularity at the Newman scale ($r = 0$) is the division in five communities, one for each clique, and three arbitrary subgroups of

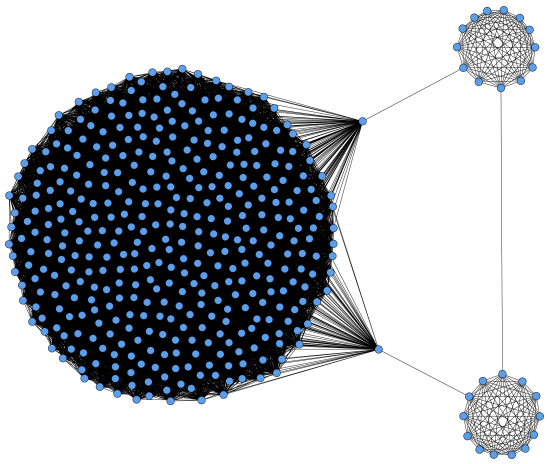


Figure 2.8: Benchmark proposed in [126] to test the resolution limit of multi-resolution methods. The large component is a ER network of 400 nodes with $\langle k \rangle = 100$ linked to two cliques of 13 nodes each, sharing only one link between them. The goal is to separate the three subgraphs using a community detection algorithm aimed to detect multiple resolutions.

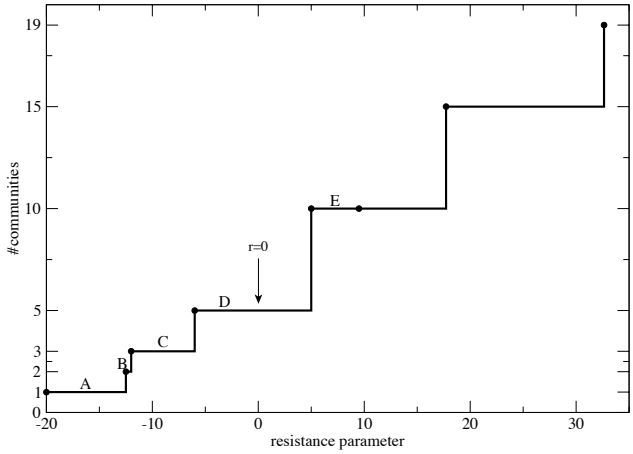



Figure 2.9: AFG mesoscales for the LF benchmark. Here we plot the partitions obtained as a function of the value of the resolution parameter used. We do find a subdivision of the network in three, but it does not correspond to the desired partitioning.

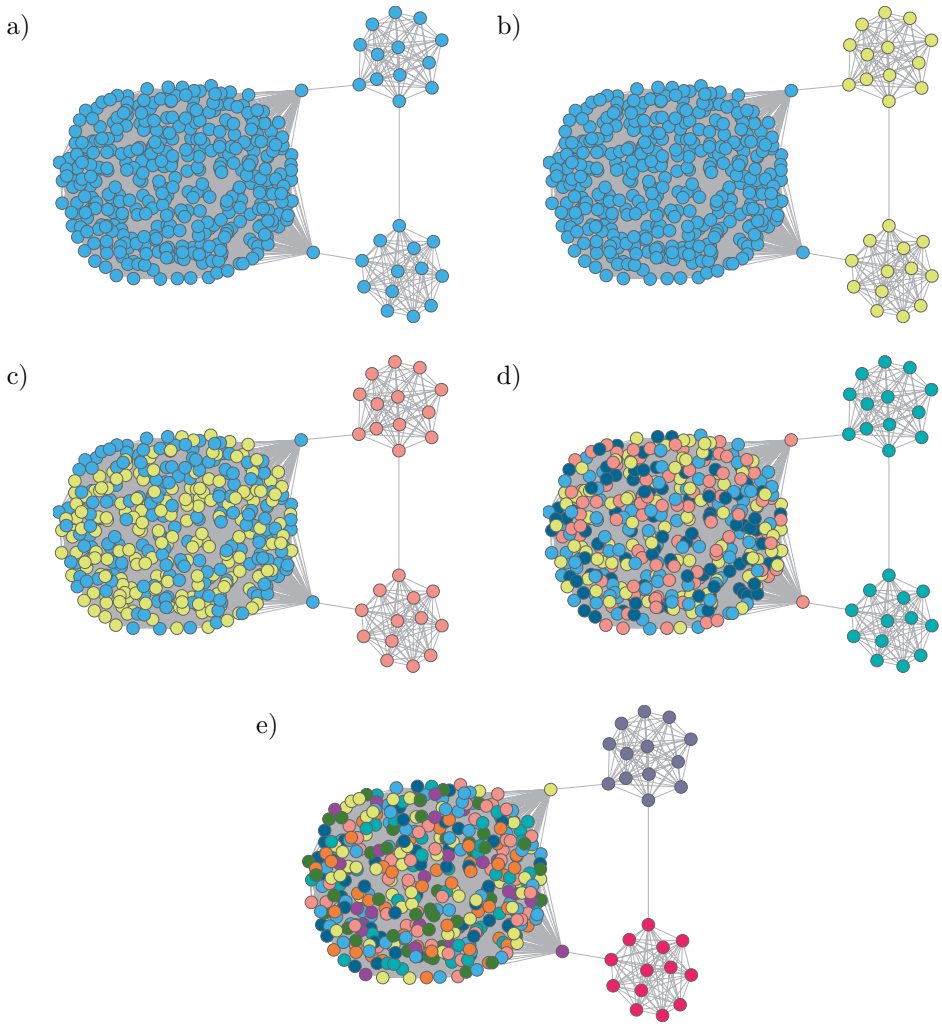



Figure 2.10: Plot of the partitions of the LF benchmark according to the AFG algorithm. **a)** Macroscale. **b)** Division in two communities. **c)** and **d)** The division in 3 and 5 communities breaks the big subnetwork before separating the two cliques. **e)** When modularity is finally able to separate the two cliques, the big subgraph is broken in eight pieces.

nodes belonging to the ER network. For the case where we approach this problem using multi-resolution algorithms, tests using the AFG method show that it is not possible to obtain a partition containing the three desired communities, the reason lying in the big difference between the sizes of the groups. In Fig. 2.9, we plot the partitions obtained for a subset of the mesoscale, starting out from the macroscale (partitions of a very high number of communities are uninteresting for our purpose and therefore are not shown). The exact groupings obtained for the partitions labeled A-E are shown in Fig. 2.10. As we can see, there is no scale which delivers the desired partitioning, because when the resistance parameter is sufficiently high as to correctly detect the two cliques, this same value of the parameter causes the ER network to divide in eight groups. The main problem seems to be that a single resolution parameter is not enough, which suggests that we could assign a parameter for each substructure. This is the simple idea behind the Hierarchical AFG method, which is presented next.

### 2.3.2 HIERARCHICAL APPROACH TO SOLVE THE SPLITTING AND MERGING BEHAVIOR OF MODULARITY

The intuition behind the resistance parameter of the original AFG algorithm was to achieve the effect of a magnifying lens, so that we could observe the network from a shorter or longer distance, and optimize modularity at each resolution. Now, the benchmark proposed by Lancichinetti & Fortunato emphasizes a problem present in this approach: in some situations a single resolution parameter does not suffice. The problem is similar to that of trying to take a picture of an elephant next to an ant. If we have a camera with a single zooming lens, either we will capture the whole elephant and the ant will not be visible, or we will capture the ant and only some parts of the elephant. The solution then, relies on using the appropriate resolutions for each part of the network.

Our approach to solve the resolution problem takes advantage of the capability of the AFG method to find meaningful communities from the initial steps of the mesoscale analysis. More precisely, we propose the use of an iterative scheme which combines the optimization of modularity close to the macroscale of the network with its splitting in subgraphs, one for each of the previously found communities.

For the sake of simplicity, we will refer to undirected unsigned networks in this work. In the particular case that the original network does not have negative

weights, $r < 0$ and $w_{ii} = 0$, $\forall i$, the modularity formulation for the AFG method reads

$$\begin{aligned} Q_{\mathrm{AFG}}[w_{ij}, C, r] &= \frac{1}{2w + N|r|} \sum_i \sum_j \left[ w_{ij} + r\delta_{ij} - \left( \frac{w_i w_j}{2w} - \frac{r^2}{N|r|} \right) \right] \delta(C_i, C_j) \\ &= \frac{2w}{2w - Nr} Q[w_{ij}, C] + \frac{r}{2w - Nr} \left( N - \frac{1}{N} \sum_{s \in C} n_s^2 \right), \end{aligned} \tag{2.38}$$

In this case, the prescription of our algorithm is the following:

1. Start out from the macroscale partition $\mathcal{M}$, which has only one community containing all nodes. Then, find the upper bound of this macroscale, which is the minimum value of the resistance parameter ($r_{\min}$) needed to find a partition $\mathcal{C}$ of the network with optimal modularity $Q_{\mathrm{AFG}}[w_{ij}, \mathcal{C}, r_{\min}]$ formed by more than one community.
2. Split the network in the subgraphs defined by the partition $\mathcal{C}$ just found, effectively creating new separate networks.
3. Repeat the previous steps with each subgraph until no further subdivisions are needed.

This algorithm defines a hierarchical organization of the nodes, where the values of $r_{\min}$ at each splitting define the ultra-metric distances between nodes, i.e. the heights in the dendrogram at which every pair of nodes first meet.

The calculations of $r_{\min}$ and $\mathcal{C}$ may be performed simultaneously, therefore avoiding the costly scanning of the whole mesoscale between the lower and upper bounds of the resistance. This is a consequence of the following properties:

- The value of $r_{\min}$ is negative, with the only exception in which the network is just a clique.
- $Q_{\mathrm{AFG}}[w_{ij}, \mathcal{M}, r] = 0$, $\forall r < 0$, because:

$$\begin{aligned} Q_{\mathrm{AFG}}[w_{ij}, \mathcal{M}, r] &= \frac{1}{2w + N|r|} \sum_i \sum_j \left[ w_{ij} + r\delta_{ij} - \left( \frac{w_i w_j}{2w} - \frac{r^2}{N|r|} \right) \right] \\ &= \frac{1}{2w + N|r|} \left[ 2w + Nr - \left( \frac{(2w)^2}{2w} + \frac{N^2 r^2}{Nr} \right) \right] = 0\,. \end{aligned} \tag{2.39}$$

In fact, modularity Eq. 2.17 is always zero for $\mathcal{M}$, no matter the network or the value of the self-loops.

- Since $Q_{\text{AFG}}[w_{ij}, \mathcal{M}, r] = 0$ and modularity is a continuous and monotonically increasing function of the resistance for any given $C \neq \mathcal{M}$, the optimal partition $\mathcal{C}$ at $r_{\min}$ must satisfy $Q_{\text{AFG}}[w_{ij}, \mathcal{C}, r_{\min}] = 0$.
- For any given partition $C$, the minimum meaningful value of the resistance $r_{\min}(C)$ is the one for which $Q_{\text{AFG}}[w_{ij}, C, r_{\min}(C)] = 0$. Thus, Eq. 2.38 leads to

$$r_{\min}(C) = \frac{-2w}{N - \frac{1}{N}\sum_{s \in C} n_s^2} Q[w_{ij}, C]\,. \tag{2.40}$$

- The upper bound of the macroscale is given by

$$r_{\min} = \min_{C} \{r_{\min}(C)\} \tag{2.41}$$

and $\mathcal{C}$ is the partition which minimizes $r_{\min}(C)$.

All these properties may be combined in the following *fast-tracking resistance* (FTR) algorithm to find the upper bound of the macroscale:

1. Optimize modularity at $r = 0$, to obtain partition $C_{\text{prev}}$.
2. Calculate $r_{\min}(C_{\text{prev}})$ using Eq. 2.40.
3. Optimize modularity at $r := r_{\min}(C_{\text{prev}})$, to obtain the current partition $C_{\text{curr}}$.
4. If $C_{\text{curr}} = C_{\text{prev}}$ or $C_{\text{curr}} = \mathcal{M}$, then $r_{\min} := r_{\min}(C_{\text{prev}})$ and $\mathcal{C} := C_{\text{prev}}$.
5. Otherwise, let $C_{\text{prev}} := C_{\text{curr}}$ and go back to the second step.

In practice, this algorithm converges in a few number of steps. It stops when a value of $r$ is found such that the optimization of modularity does not produce any new partition. In this case, the modularity of both $C_{\text{prev}}$ and $\mathcal{M}$ is zero, and no known partition can be used to obtain a better upper bound of the macroscale. Of course, we cannot claim that we have found the "real" $r_{\min}$, since no optimization heuristic can ensure the finding of the global maximum of modularity, but this is the best approximation one may obtain. To exemplify the functioning of the FTR algorithm we show in Fig. 2.11 its application to the first hierarchical splitting of Zachary Karate Club network [221].

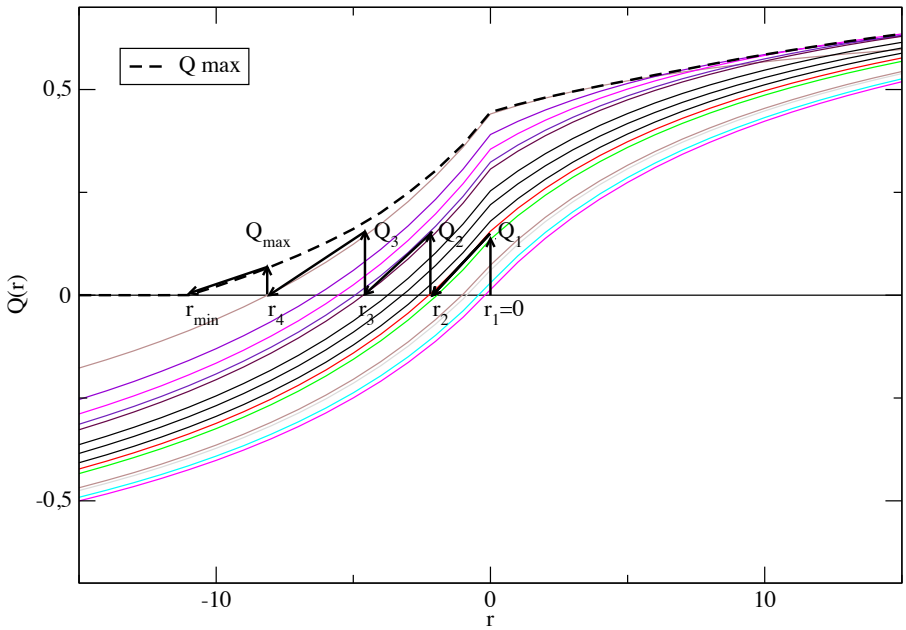



Figure 2.11: Example of evolution of the FTR algorithm finding $r_{\min}$ in four iterations of the scheme. We start at $r_1 = 0$, optimizing modularity we find $Q_1$, we look for the $r_{\min}$ corresponding to the partition found at $Q_1$ using Eq. 2.40 and label it $r_2$, the process follows with $Q_2 \to r_3 \to Q_3 \to r_4 \to Q_{\max}$ up to finding $r_{\min}$, beyond this value the only partition we will find corresponds to the whole network as a unique module. Different curves in color are values of $Q[w_{ij}, \mathcal{C}, r]$ for different partitions.

To illustrate the performance of the hierarchical scheme proposed, we have applied it to the LF benchmark introduced previously. We use the FTR algorithm to speed up the process of finding the minimal $r$ at which every subgraph splits. The aim is to find the partition in three communities in which the giant ER and each clique are separated. These three communities should contain the nodes labeled 1 to 400, 401 to 413 and 414 to 426, respectively.

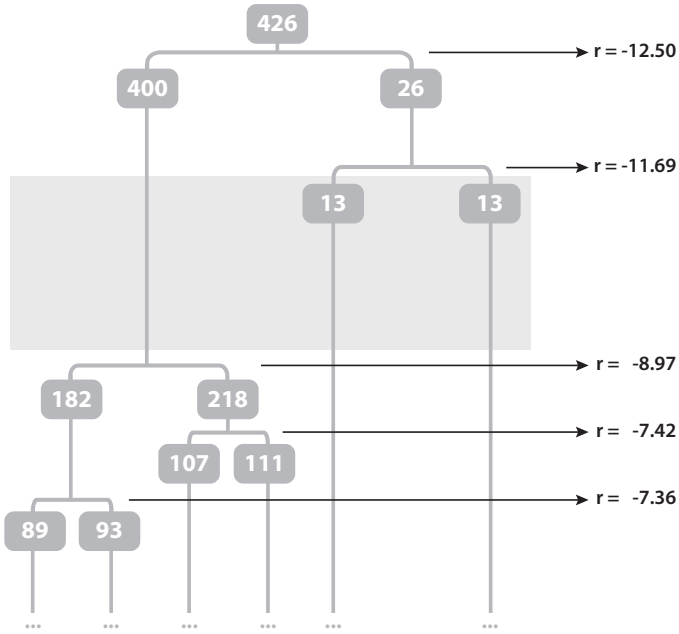



Figure 2.12: Dendrogram resulting from the application of the hierarchical multi-resolution method on the benchmark of Fig. 2.8. The grey region shows the range of the resistance parameter in which the three communities searched coexist. Note that the vertical lines are not scaled.

As stated in the method, we have started out from the macroscale $\mathcal{M}$ of the network, which contains the 426 nodes. The optimal partition splits in two communities at a value of the resistance parameter -12.5, obtaining a community formed by the nodes from 1 to 400 and another community containing the 26 nodes corresponding to the two cliques. Performing the hierarchical method on the two communities obtained, we find that the community containing the 26 nodes easily splits in two communities of 13 nodes, at a value of the resistance equal to

-11.69. The partition containing 400 nodes splits in two at a greater value of the resistance parameter, which is -8.97. After that, a hierarchical multi-resolution is applied to any community found, until no further divisions are needed. The results of this example are shown in a dendrogram representation in Fig. 2.12.

Observing this figure, we find that there is a region of the resistance parameter in which the three communities we were hoping to find coexist. This happens because the two cliques form their own communities much before the community of 400 nodes is split in two. As in any algorithm which delivers a hierarchy of partitions, we have the problem of choosing which is the horizontal cut of the dendrogram that we are going to take as our solution. We propose to make use of the values of the resistance parameter at each split of the dendrogram, in a way equivalent to the idea of the AFG algorithm, where we kept those partitions that remained unchanged for more values of $r$. If we observe the dendrogram, we will see that the grey region, which is the one where the three desired communities coexist, is also very stable in terms of values of the resistance. We propose that, for the case where the communities are not known beforehand, it is appropriate to do a complete hierarchical screening and choose the horizontal cut corresponding to the longest range of values of $r$.

Note that the result obtained using the hierarchical approach cannot be obtained using the original AFG method neither with the original formulation of modularity. The rationale behind the success of the hierarchical method in this situation is the following: the separation of the network in optimal subgraphs, each one split and independently analyzed through the multi-resolution scheme, reduces the global resolution limit. This resolution limit depends on the number of nodes and the number of links in the whole structure. The multi-resolution method is able to focus the attention on lower scales while other parts of the network are being screened independently at larger resolution values of $r$.

## 2.4 BENCHMARK MODEL FOR GENERATING DYNAMIC COMMUNITIES

Real life networked data is often obtained from the observation of dynamical processes. Up to now, we have focused in representing such data in simple single layer networks, meaning that we merely account for the interactions between elements, without any tag on the timestamp they may have. However, most dynamical pro-

cesses evolve in time, take for instance one example of social dynamics: a network of friendship relations. Human relations change through time, new entities are added and new connections are created, and the opposite also happens. While sometimes we may only have access or be interested in a single static picture of the system, it is very interesting to capture several snapshots through time and observe its evolution. If we have access to such valuable information, it is important to respect the temporal nature of this data, and to represent it accordingly. This is usually done by means of a *time-varying* or *temporal network*, a multidimensional network consisting of one layer per each timestamp, where each layer is a static network describing the observation at that time. Usually, nodes in one layer are connected with their counterparts in the next layer with a directed link, to account for the directionality of time. In brief, time-varying networks can be understood as a particular case of multiplex networks with directed inter-layer links. Regarding the study of such networks, the same amount of questions that we might pose to single layer networks can be translated to time-varying networks, with the added complexity of handling a multidimensional structure. In particular, a very interesting feature of time-varying networks is the structure of communities. In the particular example of a social network, access to the mesoscopic structure would be informative of how people organize into groups, and how these groups grow, shrink, split or merge with other groups, as time goes by. However, the problem of unraveling the evolution of the community structure is even more delicate than its single layer counterpart, due to the array of possible additional definitions that the concept of *evolving community* may have. Furthermore, very few temporal data with a known community structure is available, which hinders the task of assessing the performance of any algorithm devoted to detect evolving communities. Next, I will introduce a work aimed to help in this task, by building a benchmark model for dynamic communities.

### 2.4.1 EVOLVING COMMUNITIES IN TIME-VARYING NETWORKS

The analysis and modeling of temporal networks has received a lot of attention lately, mainly due to the increasing availability of time-stamped network datasets [123, 107, 172, 202, 14]. A relevant issue is whether and how the community structure of networks [77] changes in time. Communities reveal how networks are organized and function, hence major changes in their configuration

might signal important turns in the evolution of the system as a whole, possibly anticipating dramatic developments such as, e.g. rapid growth or disruption.

Indeed, there has been a lot of activity around this topic over the past years [38, 167, 182, 146, 96, 16, 33, 51]. However, most investigations lack strength on the validation part, which typically consists in checking whether the results of the algorithm "make sense" in one or more real networks whose community structure is usually unknown. Actually, it is not obvious what exactly does it mean to test an algorithm for detecting evolving communities. One idea could be that of correctly identifying the community structure of the system at each time stamp. However, during the evolution of the system several events that affect the network structure may occur, such as the creation/deletion of nodes/links or link rewiring, and it is not possible to detect these events by observing a single time-stamped network, they require taking into account the whole picture to be properly understood.

To explicitly keep track of the history of the system, an option is to consider multiple snapshots at once. For instance, in the evolutionary clustering approach [38] the goal is to find a partition that is descriptive of the structure of a given snapshot as well as correlated to the structure of the previous snapshots. Furthermore, the added value of any approach should be the ability to promptly detect changes in the community structure of the network. It would be possible to verify this if there were suitable benchmark graphs with evolving clusters, which is exactly the purpose of the work I present here. In the following, I introduce a model, derived from the classic stochastic block models [105, 85, 128, 100], that generates three classes of dynamic benchmark graphs. The objective is to provide with time-evolving networks, such that at each snapshot the partition into communities is well defined, according to the model. To keep things simple, periodic evolution is considered, such that the same history repeats itself in cycles and is invariant under time reversal. The analysis of the community structure evolution for the designed benchmarks reveals that approaches exploiting the flow of system configurations might be more accurate in detecting the evolving community structure than methods that consider the snapshots independently.

### 2.4.2 BENCHMARK MODEL OF EVOLVING COMMUNITY STRUCTURE

The model for generating networks with evolving community structure we propose is based on the classic stochastic block model (SBM) [105], and its goal is to generate communities that have a periodic evolution. It works as follows: a

network is divided in a number $q$ of subgraphs (which we are going to take as our *planted communities*) and the nodes of the same subgraph are linked with a probability $p_{\text{in}}$, whereas nodes of different subgraphs are linked with a probability $p_{\text{out}}$. Such probabilities match the link densities within and between subgraphs. Supposing subgraphs of equal size, if $p_{\text{in}} > (q-1)p_{\text{out}}$ the resulting subgraphs are considered communities, as the (expected) link density within subgraphs exceeds their connectivity to the rest of the graph. The generation of samples from this model has a built-in efficiency: if there are $m_{\max}$ pairs of nodes, the actual number of edges is drawn from a binomial distribution with parameters $m_{\max}$ and $p$. Then, we simply place this amount of edges randomly to generate a sample from our ensemble.

The model implements the two fundamental classes of dynamic processes: the growing/shrinking and merging/splitting of communities. By combining these two reversible types of processes one can capture the most common behaviors of dynamic communities in real systems. We are then able to generate three standardized benchmarks: one consists in communities which grow and shrink in size (keeping fixed the total number of nodes of the network), while the second considers communities that split and merge. The third one is a mixed version of the previous two, which consists of a combination of the last four operations.

*Grow/shrink benchmark*

This process models the movement of nodes from one community to another. At all times, two communities are kept in a SBM ensemble with intra-community link density $p_{\text{in}}$ and inter-community link density $p_{\text{out}}$. However, the number of nodes in each community changes over time. In the basic process, we have a total of $2n$ nodes in two communities. In the balanced state, these are split into two equally sized communities of $n$ nodes each, that we call A and B. Periodically, a fraction $f$ of nodes in community A will switch to community B, and vice versa. If we take $n_{\text{A}}$ as the size of community A, then the number of nodes in the community B is $n_{\text{B}} = 2n - n_{\text{A}}$. Then, at time $t$ the number of nodes in community A is

$$n_{\text{A}} = n - nf\,[2x(t + \tau/4) - 1] \tag{2.42}$$

with the $\tau/4$ phase factor specifying equal sized communities at $t = 0$. The function $x(t)$ is the triangular waveform:

$$x(t) = \begin{cases} 2t^* & 0 \leq t^* < 1/2 \\ 2 - 2t^* & 1/2 \leq t^* < 1 \end{cases} \tag{2.43}$$

which controls the time periodicity. The constant $\phi$ is a phase factor with $\phi = 0$ for the $q = 2$ case, and specified otherwise in the case of $q > 2$. With this formulation, we get communities of sizes $(n, n)$, $(n - nf, n + nf)$, $(n, n)$, and $(n + nf, n - nf)$ at $t/\tau \bmod 1 = 0$, $\frac{1}{4}$, $\frac{2}{4}$, and $\frac{3}{4}$ respectively. In practice, the $2n$ nodes are sorted in some arbitrary order: the first $n_{\mathrm{A}}$ nodes are put into community A, and the others into community B.

After the community sizes are decided, the edges must be placed, taking into account that it is necessary that we keep the two communities in the proper SBM ensemble with equal and independent link probability of $p_{\mathrm{in}}$ at all times. The independence of pairs provides a hint on how to do this. When a node $j$ is moved from community A to B, all the existing edges of node $j$ are removed. Then, an edge is added between $j$ and each node in the destination community B with equal and independent probability $p_{\mathrm{in}}$ and between $j$ and each node in community A with equal and independent probability $p_{\mathrm{out}}$, thus the ensemble is maintained. Conveniently, all edges can be pre-computed and stored to allow a strictly repeating process, with the state at time $t$ being identical to the state at time $t + \tau$, in analogy to the merging process. Fig. 2.13 (a) depicts a sketch of the grow/shrink benchmark for the case number of communities $q = 2$.

*Merge/split benchmark*

This process models the merging of two communities. In this setup, we have a set of $2n$ nodes, divided into two communities of $n$ nodes each. Each of the two initial communities has an internal link density of $p_{\mathrm{in}}$, where those links are placed at initialization and kept unmodified over time. The two extreme cases of the periodic evolution are the unmerged and the merged state. In the unmerged state, all possible pairs of nodes between the two communities have an edge with probability $p_{\mathrm{out}}$. This means the network still has a connected component, but the nodes form two communities. In the merged state, all possible pairs of nodes between these two communities have an edge with probability $p_{\mathrm{in}}$, which implies that all pairs of nodes in the network have the same link density $p_{\mathrm{in}}$, the previous two communities are now indistinguishable, and thus we have one large community with $2n$ nodes.

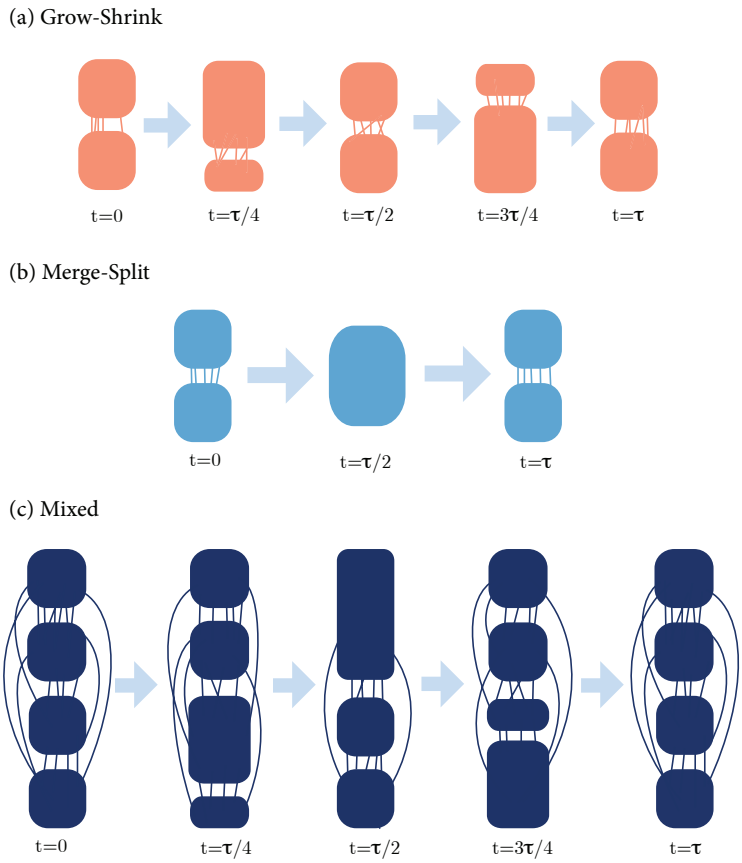



Figure 2.13: Schematic representation of the three main processes of the dynamic benchmark. **(a)** Grow/Shrink process with $q = 2$. We begin with two equal-sized communities, and over a period of $\tau$ nodes move from the bottom community to the top, then from the top to bottom, then back to the symmetric state. **(b)** Merge/Split benchmark with $q = 2$. We begin with two communities, and over a period of $\tau$ we linearly add edges until there is one community with uniform link density, then reverse the process. **(c)** Mixed benchmark with $q = 4$, combining the merging and growing processes.

The whole merge/split process is a periodic interpolation of the merged and unmerged states. The numbers of inter-community edges in the unmerged state, $m_{\mathrm{um}}$, and in the merged state, $m_{\mathrm{m}}$, are first picked from a binomial distribution consistent with the binomial distribution parameters $n^2$ and $p_{\mathrm{out}}$ or $p_{\mathrm{in}}$. All possible inter-community edges are placed in random order, and the first

$$m^*(t) = (1 - x(t))m_{\mathrm{um}} + x(t)m_{\mathrm{m}} \tag{2.44}$$

edges are selected to be active at time $t$. The effective intra-community link density is $p^*_{\mathrm{inter}}(t) = m^*(t)/n^2$. The parameter $x(t)$ is the triangular waveform from Eq. 2.43. In practice, this means at time $t/\tau \bmod 1 = 0$, the communities are unmerged, and at $t/\tau \bmod 1 = 1/2$, the communities are merged, with linear interpolation (of the number of edges) between these points. Since the possible edges are ordered only at initialization, the process is strictly periodic, that is, the edges present at time $t$ are identical to those present at time $t + \tau$.

One may think that the communities are fully merged at the extreme of this process, where the inter-community link density is $p^*_{\mathrm{inter}} = p_{\mathrm{in}}$ (at $t = \tau/2$). However, due to the *detectability limit* of communities in stochastic block models [59], this is not the case. Even when $p_{\mathrm{out}} < p_{\mathrm{in}}$, it can be that the configuration is indistinguishable from one large community. The merge/split benchmark for the configuration in $q = 2$ communities is shown in Fig. 2.13 (b).

*Mixed benchmark*

This process is a combination of the merging and growing processes. In this process, there are a total of $4n$ nodes with two merging/splitting communities ($2n$ nodes) and two growing/shrinking communities ($2n$ nodes). The intra-community links are managed with the same processes as above with phase factors of $\phi = 0$ for both. If there are $q = 4a > 4$ total communities, then the pairs of communities involved in merging and growing process have phase factors $\phi = 0, \frac{1}{a}, \frac{2}{a}, ... \frac{(a-1)}{a}$. Between the pairs of nodes that belong to different processes, an edge exists with a probability of $p_{\mathrm{out}}$. Fig. 2.13(c) pictures a sketch of the mixed benchmark for $q = 4$.

### 2.4.3 TIME-DEPENDENT COMPARISON MEASURES

The assessment of the performance of any clustering algorithm requires the use of measures to define the distance or similarity between any pair of partitions. The list of available measures is long, including e.g. the Jaccard index [111], the Rand index [178], the adjusted Rand index [108], the normalized mutual information [204], the van Dongen metric [62] and the normalized variation of information metric [141]. However, all those measures are suited for the comparison of static partitions. For this reason, we need to generalise any desired measure to the case of evolving communities (i.e. allowing the comparison between *sets* of partitions).

All the previous measures have in common the possibility of being expressed in terms of the elements of the so-called *confusion matrix* or *contingency table*, thus we focus first on its calculation. Let $\mathcal{C} = \{C_\alpha | \alpha = 1, \dots, r\}$ and $\mathcal{C}' = \{C'_{\alpha'} | \alpha' = 1, \dots, r'\}$ be two partitions of the data in $r$ and $r'$ disjoint clusters. The $\alpha\alpha'$th component of the contingency table $M$ accounts for the number of elements in the intersection of clusters $C_\alpha$ and $C'_{\alpha'}$,

$$m_{\alpha\alpha'} = |C_\alpha \cap C'_{\alpha'}| \,. \tag{2.45}$$

The sizes of the clusters simply read as $n_\alpha = |C_\alpha| = \sum_{\alpha'} m_{\alpha\alpha'}$ and $n'_{\alpha'} = |C'_{\alpha'}| = \sum_\alpha m_{\alpha\alpha'}$, and the total number of elements is $N = \sum_\alpha n_\alpha = \sum_{\alpha'} n'_{\alpha'} = \sum_\alpha \sum_{\alpha'} m_{\alpha\alpha'}$.

With these definitions at hand, one can calculate the Jaccard index,

$$J = \frac{\sum_\alpha \sum_{\alpha'} \binom{m_{\alpha\alpha'}}{2}}{\sum_\alpha \binom{n_\alpha}{2} + \sum_{\alpha'} \binom{n'_{\alpha'}}{2} - \sum_\alpha \sum_{\alpha'} \binom{m_{\alpha\alpha'}}{2}} \,, \tag{2.46}$$

the normalized mutual information index,

$$NMI = \frac{-2 \sum_\alpha \sum_{\alpha'} m_{\alpha\alpha'} \log \frac{N m_{\alpha\alpha'}}{n_\alpha n'_{\alpha'}}}{\sum_\alpha n_\alpha \log \frac{n_\alpha}{N} + \sum_{\alpha'} n'_{\alpha'} \log \frac{n'_{\alpha'}}{N}} \,, \tag{2.47}$$

and the normalized variation of information metric,

$$NVI = \frac{-1}{\log N} \sum_{\alpha} \sum_{\alpha'} \frac{m_{\alpha\alpha'}}{N} \log \frac{(m_{\alpha\alpha'})^2}{n_\alpha n'_{\alpha'}} , \tag{2.48}$$

where, by convention, $0 \log 0 = 0$.

In the case of evolving networks we have to compare two sequences of partitions, $\{\mathcal{C}(t)|t = 1, \ldots, T\}$ and $\{\mathcal{C}'(t)|t = 1, \ldots, T\}$, a task which can be performed in different ways. The simplest solution is the independent comparison of partitions at each time step, by measuring the similarity or distance between $\mathcal{C}(t)$ and $\mathcal{C}'(t)$ for each value of $t$, thus obtaining e.g. a Jaccard index $J(t)$ for each snapshot, see Fig. 2.14 (a). However, this procedure discards the evolutionary nature of the communities: we would like to evaluate not only the static resemblance of the communities but also if they evolve in a similar way.

The proposal here consists in the definition of *windowed* forms of the different indices and metrics, obtained by considering sequences of consecutive partitions, i.e. time windows of a predefined duration $\sigma$. In Fig. 2.14 (b) we show the comparison between individual snapshots and sequences of length two. For example, let us consider the time window formed by time steps from $t$ to $t + \sigma$. Every node belongs to a different cluster at each snapshot, and this evolution can be identified as one of the items in $\mathcal{D}(t;\sigma) = \mathcal{C}(t) \times \mathcal{C}(t+1) \times \cdots \times \mathcal{C}(t+\sigma)$ for the first sequence of partitions, and $\mathcal{D}'(t;\sigma) = \mathcal{C}'(t) \times \cdots \times \mathcal{C}'(t+\sigma)$ for the second one, where the cross sign denotes the cartesian product of sets. Since the number of nodes is $N$, there are at most $N$ different non-void sets $D_\alpha(t;\sigma) \in \mathcal{D}(t;\sigma)$, and also for $D'_{\alpha'}(t;\sigma) \in \mathcal{D}'(t;\sigma)$. For example, in Fig. 2.14 (b), the combinations of partitions (excluding empty sets) are $\mathcal{D}(t = 1;\sigma = 1) = \{\text{AA}, \text{AB}, \text{BB}, \text{CC}\}$ and $\mathcal{D}'(t = 1;\sigma = 1) = \{\text{AA}, \text{BB}, \text{CC}\}$. Next, we may define the elements of the contingency table for this time window as

$$m_{\alpha\alpha'}(t;\sigma) = |D_\alpha(t;\sigma) \cap D'_{\alpha'}(t;\sigma)| , \tag{2.49}$$

which accounts for the number of nodes following the same cluster evolutions $D_\alpha(t;\sigma)$ and $D'_{\alpha'}(t;\sigma)$.

Likewise, we have:

$$n_\alpha(t;\sigma) = |D_\alpha(t;\sigma)| = \sum_{\alpha'} m_{\alpha\alpha'}(t;\sigma)\,, \tag{2.50}$$

$$n'_{\alpha'}(t;\sigma) = |D'_{\alpha'}(t;\sigma)| = \sum_{\alpha} m_{\alpha\alpha'}(t;\sigma)\,, \tag{2.51}$$

and

$$N = \sum_\alpha n_\alpha(t;\sigma) = \sum_{\alpha'} n'_{\alpha'}(t;\sigma) = \sum_\alpha \sum_{\alpha'} m_{\alpha\alpha'}(t;\sigma)\,. \tag{2.52}$$

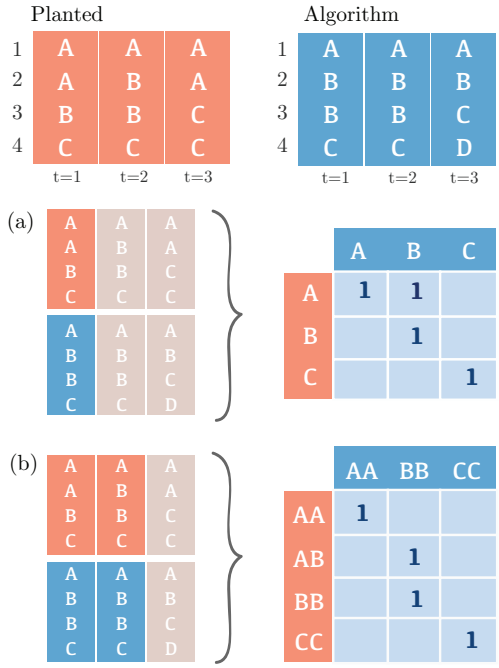



Figure 2.14: Example of construction of the contingency tables $m_{\alpha\alpha'}$. On the top we represent the time evolution of a toy network of four nodes (rows) in three time steps (columns), and the partitions in communities we want to compare, e.g. the planted partitions from the benchmark and those obtained by a certain algorithm. To compare these two partitionings, we can do it as it is depicted in **(a)**, which takes only one snapshot at a time ($\sigma = 0$), or as in **(b)**, building a contingency table where the entries consider two snapshots at the same time ($\sigma = 1$). Afterwards, the measures (NVI, NMI or Jaccard index) are calculated from these tables.

Finally, we may use Eqs. 2.46 to 2.48 to calculate the corresponding windowed Jaccard index $J(t;\sigma)$, windowed normalized mutual information index $NMI(t;\sigma)$ and windowed normalized variation of information metric $NVI(t;\sigma)$, respectively. Of course, the windowed measures reduce to the standard static ones when $\sigma = 0$, and are able to capture differences in the evolution of communities which cannot be distinguished using their classical versions. Additionally, it is very convenient to be able to quantify the overall deviation with a single number. A simple solution is the use of the average squared errors, which are expressed as follows:

$$E_{\mathrm{J}}(\sigma) = \frac{1}{T}\sum_{t=1}^{T}(J(t;\sigma)-1)^2\,, \tag{2.53}$$

$$E_{NMI}(\sigma) = \frac{1}{T}\sum_{t=1}^{T}(NMI(t;\sigma)-1)^2\,, \tag{2.54}$$

$$E_{NVI}(\sigma) = \frac{1}{T}\sum_{t=1}^{T}NVI(t;\sigma)^2\,. \tag{2.55}$$

### 2.4.4 APPLICATION OF A MULTILAYER COMMUNITY DETECTION ALGORITHM TO THE GENERATED BENCHMARK

Here we show an example of the application of a community detection algorithm, designed to take into account the evolution of complex networks, to reveal the community structure in our benchmarks. The chosen method is the multislice algorithm by Mucha et al. introduced in [146], which extends the definition of modularity to multilayer networks. In their representation, each layer (or slice) consists of a single network at a particular time. The slices are connected between them by joining each node with its counterpart in the next and previous layer, and this link has a specified weight $\omega$, equal for all links of this kind, which acts as a tuning parameter. For $\omega = 0$, no connection between slices is considered, and the algorithm is performed statically. As this value increases, more consideration is given to the communities across layers. The formulation includes an additional parameter $\gamma$, which accounts for the tuning of the resolution at which communities are found, as in [180]. For the comparison, we have used the code available in [116], setting the resolution parameter $\gamma$ to one (recovering the Newman scale of modularity) and varying the inter-slice coupling $\omega$.

The benchmarks used to put to test this algorithm are generated using the model just presented. For the sake of simplicity, we generate three simple standard benchmarks, one for each basic procedure: *grow/shrink*, *merge/split* and *mixed*. The *grow/shrink* benchmark consists in a network with $q = 2$ communities, where each community has initially $n = 32$ nodes (therefore the total size of the network is $N = 64$), with $p_{\text{in}} = 0.5$, $p_{\text{out}} = 0.05$, and $f = 0.5$, $\tau = 100$ time steps. The *merge/split* test has a variable number of communities, in this paper we use the parameters $q = 2$ communities of size $n = 32$ each, with $p_{\text{in}} = 0.5$, $p_{\text{out}} = 0.05$, and $\tau = 100$. The mixed benchmark, a combination of the previous two, has $q = 4$ communities of $n = 32$ nodes each, and the other parameters are set as in the previous cases.

In Fig. 2.15 we show the planted partitions for the three benchmarks and the results from the multislice algorithm at three different inter-slice couplings: in the extreme case $\omega = 0$ slices are considered independently, $\omega = 0.5$ is an intermediate value which provides good results, and $\omega = 2$ provides an example of the partitioning obtained when using strong coupling between layers. We have also compared the multislice method with a temporal stability approach [173], and the results obtained are very similar to the results of the multislice algorithm obtained at $\omega = 0.5$.

To quantitatively evaluate the results, we use the windowed measures introduced previously. We calculate the measures between the partitions obtained by the algorithm and the planted ones, for three values of the time window. When the time window is of size one ($\sigma = 0$), each snapshot is considered independently, that is, we have computed the measure between the planted partition at $t$ and the algorithm's result at $t$, repeating this process until $t = \tau$. Instead, with the time window of size 2 ($\sigma = 1$), we evaluate the evolution of the partitions during two consecutive time steps, following the same process but comparing the planted partitions at $[t, t + 1]$ with the algorithm's results at $[t, t + 1]$. This formulation is more restrictive, as we impose, in addition to the condition that the nodes must belong to the same community, that their evolution during two consecutive time steps is also the same. Similarly, we have also analyzed time windows of size 5 ($\sigma = 4$) to check the quality of the detected community evolutions at longer ranges.

Fig. 2.16 shows the results for the normalized variation of information (NVI). In general, we observe that the performance of the algorithm when using values of the coupling parameter $\omega = 2$ are better than those using $\omega = 0$ or $\omega = 0.5$.

This holds true for the *grow/shrink* case, where we see that for $\omega = 2$ the NVI is approximately zero for all values of the time window; and also for the *merge/split* benchmark, where the NVI for $\omega = 2$ is always lower or equal to the other two. What we also observe in these two benchmarks is that considering time windows larger than one (thus evaluating the evolution of the communities), the high values of NVI are exaggerated, as we can see if we compare between rows of the same column. Opposite to the other two benchmarks, the *mixed* case remains quite unchanged, obtaining small differences of the NVI values for different coupling parameters and time window sizes. Finally, the NVI squared errors reported in Table 2.17 and calculated using Eq. 2.55 are in perfect agreement with this analysis. The results using NMI and Jaccard indices also support these observations. Thus, we may conclude that, in this case, the use of memory to track the evolution of communities is convenient, but the trade-off between the continuity of the community structure and its static relevance must be carefully adjusted.

Summing up, the simple model based on SBM that has been introduced allows the construction of time-dependent networks with evolving community structure. It is useful for benchmarking purposes in testing the ability of community detection algorithms to track properly the structural evolution of networks. Also, the extended time-dependent measures for the comparison of different partitions in the dynamic case allow for the observation of differences between the outcome of the algorithms and the planted partitions through time. The code for benchmark generation and time-dependent comparison indices is available at [50] and released under the GNU General Public License.

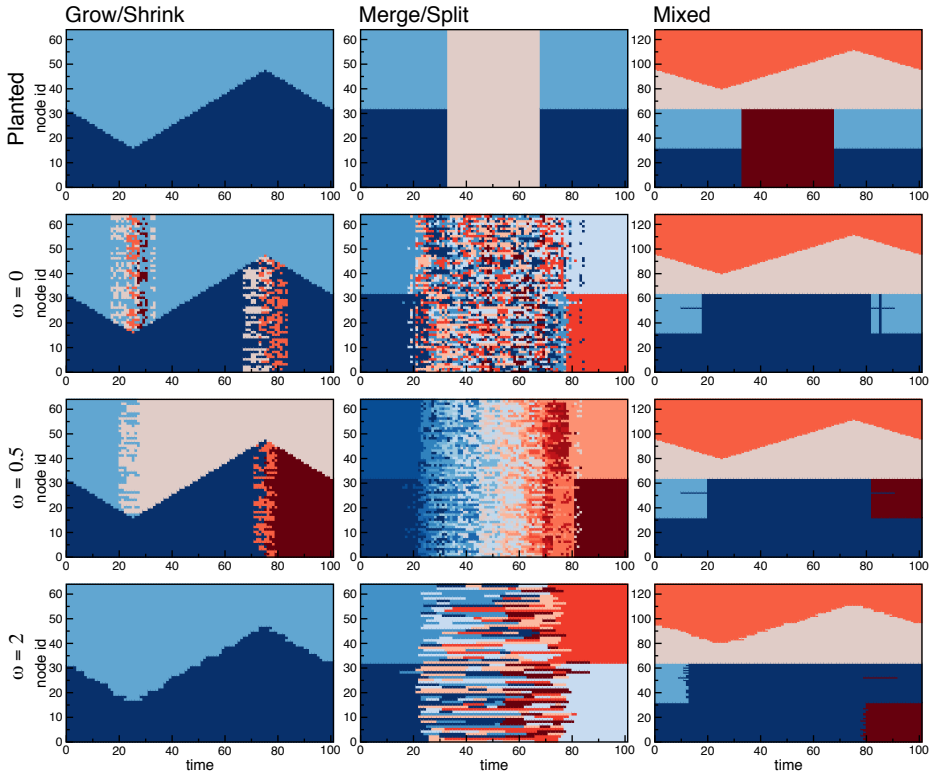



Figure 2.15: Results of the application of the multislice community detection method to the three benchmarks proposed (in columns). The first row corresponds to the planted partition of each benchmark, while the three remaining rows are the partitions obtained by the multislice algorithm for different values of the inter-slice parameter $\omega$, which is the weight of the coupling between different instances of the same nodes across layers. When $\omega = 0$ the slices are disconnected and the community detection analysis is done for each slice separately. As this value increases, more importance is given to the evolving nature of the problem, and communities across slices are found. In each plot, the vertical axis corresponds to the index of nodes in the network, while the horizontal axis represents the time. The color of each pair {node, time} accounts for the label of the community at which the node is assigned at that specific time.

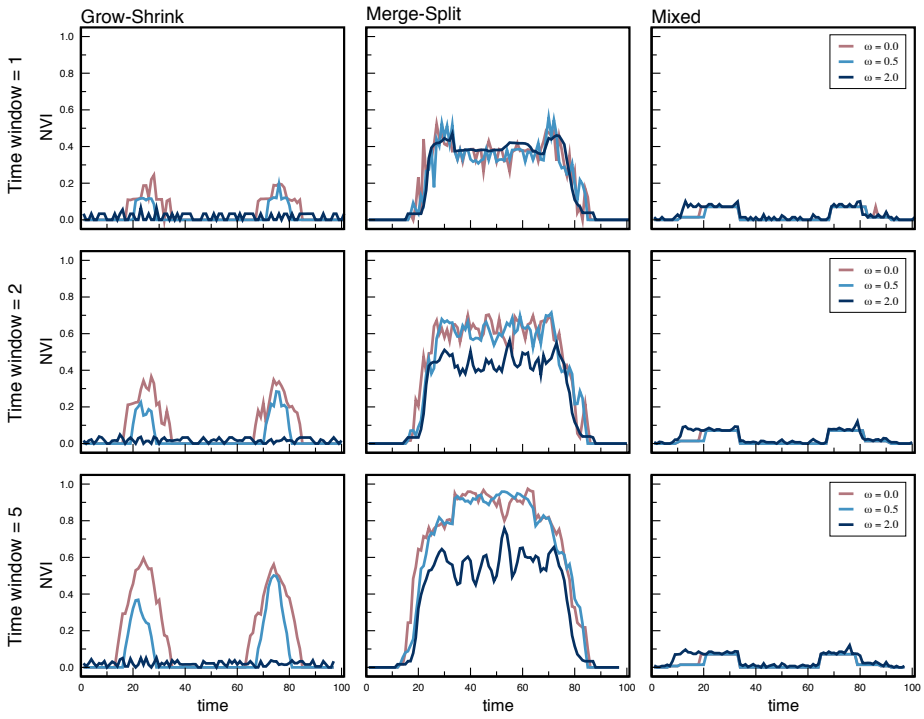



Figure 2.16: Plots of the normalized variation of information (NVI) between the planted partition and the results of the multislice algorithm in Fig. 2.15, for three different inter-slice couplings and for the three benchmarks proposed. NVI is computed using the proposed evolving formulation and for three different window sizes: 1, 2 and 5. There is a column for each benchmark, and a row for each time window size.

| | Time window | NVI squared error Grow/Shrink | Merge/Split | Mixed |
|---|---|---|---|---|
| Multislice $\omega = 0.0$ | 1 | 0.0065 | 0.0851 | 0.0015 |
| | 2 | 0.0201 | 0.2146 | 0.0015 |
| | 5 | 0.0658 | 0.4427 | 0.0016 |
| Multislice $\omega = 0.5$ | 1 | 0.0023 | 0.0808 | 0.0014 |
| | 2 | 0.0067 | 0.2019 | 0.0014 |
| | 5 | 0.0242 | 0.4278 | 0.0015 |
| Multislice $\omega = 2.0$ | 1 | 0.0006 | 0.0878 | 0.0023 |
| | 2 | 0.0005 | 0.1113 | 0.0024 |
| | 5 | 0.0006 | 0.1922 | 0.0029 |

Figure 2.17: Table of the NVI squared errors, for each method tested and each benchmark in Fig. 2.15, considering three different time windows.

UNIVERSITAT ROVIRA I VIRGILI
FROM COMMUNITY STRUCTURE TO THE PHYSICS OF MULTIPLEX NETWORKS
Clara Granell Martorell

# 3

# ON SPREADING DYNAMICS ON MULTIPLEX NETWORKS

Recently, the complex networks community has turned their attention to multilayer networks. Such networks, specially suited to represent systems with more than one type of interaction or differently tagged entities, are an appropriate generalization of single-layer networks. The power of representing such systems in a way that respects the multilevel nature of the data is twofold: not only we are able to do the right measurements of the structure of those systems, but we can also explore the effect of dynamics in such rich environments. A particularly interesting setup comes from those multilayer interconnected networks in which the nodes represent the same entities in all layers, known as *multiplex networks*. In recent literature, multiplex networks have been the substratum of multiple studies: in [125], they are used to represent transportation systems, formed by two layers accounting for the physical infrastructure and traffic flows. In a social environment, multiplex networks are used in [206] to represent multiple types of interactions in a massive multiplayer online game, where connections between users can be friendship, communication or trade ties, among others. In [186], a study for the interactions between the spreading of two diseases with different propagation mechanisms is presented. Also, in [95], the authors study evolutionary game theory when the same agents share different environments and are allowed to take different strategies in each of them. These are just a few of a big amount of works devoted to study the effect of systems comprising multiple interactions. In the following, I will present a work devoted to study the interplay between two spreading processes that compete with each other: epidemics and information spreading.

## 3.1 INTERPLAY BETWEEN INFORMATION AND EPIDEMIC SPREADING ON MULTIPLEX NETWORKS

Dynamical processes in real world are not isolated but they interact each other. Epidemic spreading, for instance, can be effectively represented as a diffusive process in a single layer, using well-known models such as Susceptible-Infected-Susceptible (SIS), Susceptible-Infected-Recovered (SIR), and others. The networked substratum for the spreading of these processes depends on the nature of the epidemics of choice, e.g. if we are interested in modeling the spreading of flu or other air-transmitted diseases, a network representing the physical contacts of people would be an appropriate setup to use. This network would contain people who spends a certain amount of time near you, and therefore are able to infect you, such as coworkers, family or friends.

However, this particular spreading could be affected by other processes, such as diffusion of information about the epidemics itself [142]. Governmental campaigns to promote the use of condoms to prevent sexually transmitted diseases or advertisements warning of seasonal flu and recommending the population to get a flu shot are examples of information of this nature. This information is shared between people in a network that may have different connectivity patterns than the one of physical contacts, this would be the case if the person posted the information on Twitter or Facebook, for example. When there is an outbreak of an epidemics, information about the presence of this disease begins to spread in the population, and it is known that having access to this information (being *aware* of the disease) often leads to individuals changing habits and taking measures to avoid getting infected.

This scenario can be summarized as two diffusive processes that have different spreading mechanisms but that affect each other, raising the question of *how does the spreading of information affect the spreading of the epidemics*. In the following, I present a model devoted to answer the previous question, studying the behavior of these two coupled dynamics in multiplex networks.

### 3.1.1 MODEL FOR AWARENESS AND EPIDEMIC SPREADING IN MULTIPLEX NETWORKS

Here we describe a setup involving two competing spreading processes: the spreading of information holds back the spreading of the disease, while the nodes infected by the disease support the information spreading process by generating new aware individuals. The abstracted model is then as follows: consider a multiplex network formed by two layers; the bottom layer is formed by the network of physical contacts, while the top one is a representation of an online social network. All nodes represent the same entities in both layers, but the connectivity is different in each of them. On top of the physical contacts layer we assimilate a Susceptible-Infected-Susceptible (SIS) process. Similarly, on top of the virtual network we place a UAU process (which stands for Unaware-Aware-Unaware), a cyclic process equivalent to SIS that accounts for the spreading of information. A sketch of the resulting scenario is shown in Fig. 3.1.

The states in this process account for users being unaware (U), and aware (A) of the existence of the epidemics and its prevention. Unaware individuals do not know that they are at risk of getting infected by an epidemics, while aware individuals do and therefore can reduce their risk to be infected. Awareness can come from two sources: the communication with aware neighbors (therefore a node becomes aware with probability $\lambda$ when an aware neighbor contacts him) or because the individual is infected and thus we assume that he is automatically aware. Since the awareness corresponds to cycles parallel to the seasonality of the epidemics, there is a certain probability of an individual forgetting the information or not caring about it, and becoming again, at all effects, unaware (which happens spontaneously with probability $\delta$).

In the physical layer, the nodes are susceptible (S) or infected (I). The infection propagates from certain infected individuals to their neighbors with probability $\beta$ once a contact between an infected neighbor has been done, and infected nodes eventually recover with probability $\mu$. After an individual gets infected it is automatically aware of the infection and changes its state in the top contact layer. On the other hand, if an individual is aware in the virtual layer and is susceptible in the physical layer, it reduces its own infectivity by a factor $\gamma$. We distinguish between the original unaware infectivity $\beta^U$ and the subsequent infectivity after being aware of the infection $\beta^A = \gamma\beta^U$. In the particular case of $\gamma = 0$, the aware individuals are completely immune to the infection.

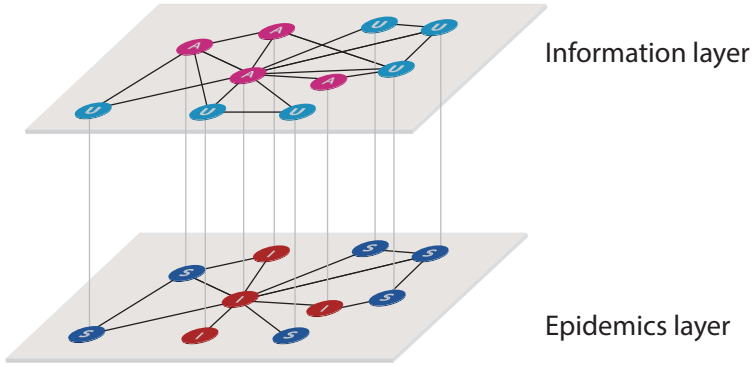



Figure 3.1: Sketch of the multiplex structure type used in this work. The upper layer (virtual contact) is supporting the spreading of awareness, where nodes have two possible states: unaware (U) or aware (A). The lower layer (physical contact) corresponds to the network where the epidemic spreading takes place. The nodes are the same actors than in the upper layer, but here their state can be: susceptible (S) or infected (I).

According to this scheme, an individual can be in three different states: unaware and susceptible (US), aware and susceptible (AS), or aware and infected (AI). Note that the state unaware and infected (UI) is spurious because according to the definition of the dynamical process stated it becomes immediately AI.

### 3.1.2 MMCA FORMULATION FOR THE UAU-SIS MODEL

A methodological way to discover the dynamical equations governing the system is to build first discrete transition probability trees that account for all the possible changes of state (and their probabilities) at every time step. For the sake of clarity, let us illustrate how to build these trees using a stand-alone SIS process, with nodes in states either Susceptible (S) or Infected (I). As explained above, $\beta$ represents the probability that a susceptible node becomes infected after a contact with one of its infected neighbors, and $\mu$ is the recovery probability for infected nodes. In a standard SIS model (reactive process [44, 86]), each infected node contacts all its neighbors at each time step, thus it is convenient to define the probability $1 - q_i$ that a susceptible node $i$ gets infected by at least one of its infected neighbors. Conversely, $q_i$ represents the probability that none of the neighbors of $i$ infects it (see Eq. 1.40). The possible changes of state of the nodes

and their probabilities at every time step can be represented by the following state transition trees:


The roots of the trees represent the possible states of a node at time $t$, hence the need of two trees, one for state I and another for state S. The leaves of each tree account for the possible states at time $t+1$. The transition arrows are labeled with the corresponding probabilities, and they depend on the node (e.g. $q_i$) and also on the time step $t$; this time dependence has not been made explicit in the trees for the sake of simplicity. From these transition trees it is possible to recover the Microscopic Markov Chain Approach (MMCA) equations [90] which express the probability of a node being in each state at time $t+1$ as a function of its state in the previous time step. For instance, the probability $p_i^{\mathrm{I}}(t+1)$ of node $i$ being infected (I) at time $t+1$ has two contributions, one for each branch in which state $I$ appears as a leaf in the trees. From the left tree we get the contribution $p_i^{\mathrm{I}}(t)(1-\mu)$, which corresponds to the case in which the node was infected (I) and has not recovered, and from the tree in the right we get $p_i^{\mathrm{S}}(t)(1-q_i(t))$, which accounts for the case in which the node was healthy (S) but has been infected by any of its neighbors. After doing the same procedure for the branches ending in state S, the resulting equations are:

$$p_i^{\mathrm{I}}(t+1) = p_i^{\mathrm{I}}(t)(1-\mu) + p_i^{\mathrm{S}}(t)(1-q_i(t))\,, \tag{3.1}$$

$$p_i^{\mathrm{S}}(t+1) = p_i^{\mathrm{I}}(t)\mu + p_i^{\mathrm{S}}(t)q_i(t)\,. \tag{3.2}$$

Which match exactly the MMCA equations introduced in Sec. 1.4.3.1. These two equations fulfill, for all time steps, the normalization condition $p_i^{\mathrm{I}} + p_i^{\mathrm{S}} = 1$, thus only one of them is really needed, which is the standard MMCA equation for the SIS process:

$$p_i^{\mathrm{I}}(t+1) = p_i^{\mathrm{I}}(t)(1-\mu) + (1-p_i^{\mathrm{I}}(t))(1-q_i(t))\,. \tag{3.3}$$

Following this procedure we can deal with more complex situations as the proposed model of competing awareness and epidemic spreading. Let us denote $a_{ij}$ and $b_{ij}$ the adjacency matrices that support the UAU and the SIS processes,

respectively. Every node $i$ has a certain probability of being in one of the three states at time $t$, denoted by $p_i^{\mathrm{AI}}(t)$, $p_i^{\mathrm{AS}}(t)$, and $p_i^{\mathrm{US}}(t)$ respectively. Assuming the absence of dynamical correlations [27], the transition probabilities for node $i$ not being informed by any neighbors $r_i(t)$, not being infected by any neighbors if $i$ was aware $q_i^{\mathrm{A}}(t)$, and not being infected by any neighbors if $i$ was unaware $q_i^{\mathrm{U}}(t)$ are

$$r_i(t) = \prod_j (1 - a_{ji} p_j^{\mathrm{A}}(t)\lambda) \tag{3.4}$$

$$q_i^{\mathrm{A}}(t) = \prod_j (1 - b_{ji} p_j^{\mathrm{AI}}(t)\beta^{\mathrm{A}}) \tag{3.5}$$

$$q_i^{\mathrm{U}}(t) = \prod_j (1 - b_{ji} p_j^{\mathrm{AI}}(t)\beta^{\mathrm{U}}) \tag{3.6}$$

where $p_j^{\mathrm{A}} = p_j^{\mathrm{AI}} + p_j^{\mathrm{AS}}$.

The transition trees for our model are shown in Fig. 3.2. They are structured according to the three phases in which every time step is divided: awareness spreading (UAU process), epidemic spreading (SIS process) and self-transmission of awareness when being infected. The resulting MMCA equations representing the probabilities of every node node being in each of the three possible states are:

$$p_i^{\mathrm{US}}(t+1) = p_i^{\mathrm{AI}}(t)\delta\mu + p_i^{\mathrm{US}}(t) r_i(t) q_i^{\mathrm{U}}(t) + p_i^{\mathrm{AS}}(t)\delta q_i^{\mathrm{U}}(t) \tag{3.7}$$

$$p_i^{\mathrm{AS}}(t+1) = p_i^{\mathrm{AI}}(t)(1-\delta)\mu + p_i^{\mathrm{US}}(1 - r_i(t)) q_i^{\mathrm{A}}(t) + p_i^{\mathrm{AS}}(t)(1-\delta) q_i^{\mathrm{A}}(t) \tag{3.8}$$

$$\begin{aligned} p_i^{\mathrm{AI}}(t+1) = {} & p_i^{\mathrm{AI}}(t)(1-\mu) + p_i^{\mathrm{US}} \left[(1 - r_i(t))(1 - q_i^{\mathrm{A}}(t)) + r_i(t)(1 - q_i^{\mathrm{U}}(t))\right] \\ & + p_i^{\mathrm{AS}}(t) \left[\delta(1 - q_i^{\mathrm{U}}(t)) + (1-\delta)(1 - q_i^{\mathrm{A}}(t))\right] \end{aligned} \tag{3.9}$$

Solving iteratively the system of Eqs. 3.7 to 3.9, together with Eqs. 3.4, 3.6 and 3.5, we can track the time evolution of the awareness and the epidemics for any initial condition. Moreover, interestingly, we can solve analytically the stationary state of the full system, and determine the onset of the epidemics as a function of the rest of the parameters of the model.

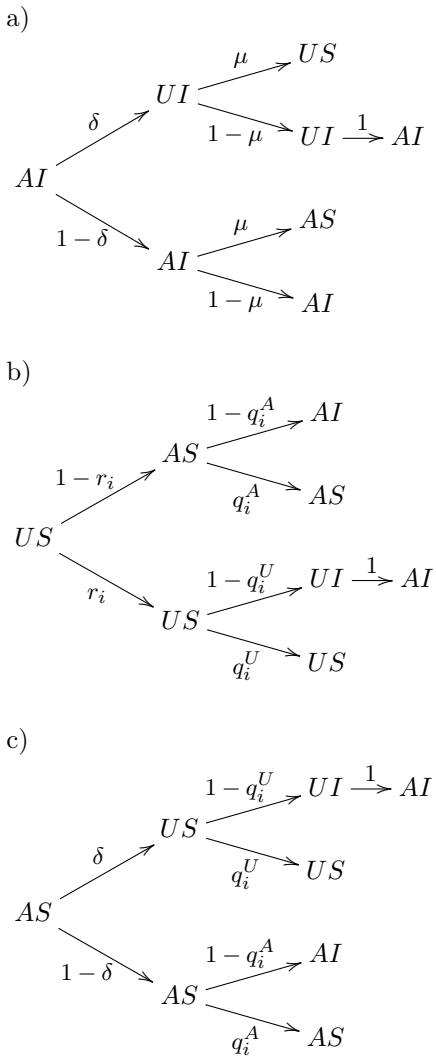



Figure 3.2: Transition probability trees for the states of the UAU-SIS dynamics in the multiplex per time step. The notation is: (AI) aware-infected, (AS) aware-susceptible, (UI) unaware-infected, (US) unaware-susceptible, $\delta$ transition probability from aware to unaware, $\mu$ transition probability from infected to susceptible, $r_i$ transition probability from unaware to aware given by neighbors, $q_i^A$ transition probability from susceptible to infected, if node is aware, given by neighbors, and $q_i^U$ transition probability from susceptible to infected, if node is unaware, given by neighbors.

### 3.1.3 DETERMINING THE ONSET OF THE EPIDEMICS

The stationary solution of the system of Eqs. 3.7 to 3.9 is computed as a set of fixed point equations satisfying $p_i^{\mathrm{AI}}(t+1) = p_i^{\mathrm{AI}}(t) = p_i^{\mathrm{AI}}$ and equivalently for (US) and (AS). Using stationarity we are now in the position of computing the onset of the epidemics $\beta_c$. Near the critical point the MMCA can be expanded assuming that the probability of nodes to be infected in the physical layer is $p_i^{\mathrm{AI}} = \epsilon_i \ll 1$. Consequently, $q_i^{\mathrm{A}} \approx 1 - \beta^{\mathrm{A}} \sum_j b_{ji}\epsilon_j$ and $q_i^{\mathrm{U}} \approx 1 - \beta^{\mathrm{U}} \sum_j b_{ji}\epsilon_j$. Inserting this in Eqs. 3.7-3.9 we obtain

$$\begin{aligned} p_i^{\mathrm{US}} &= p_i^{\mathrm{US}} r_i + p_i^{\mathrm{AS}} \delta \\ p_i^{\mathrm{AS}} &= p_i^{\mathrm{US}}(1-r_i) + p_i^{\mathrm{AS}}(1-\delta) \\ \mu\epsilon_i &= \left(p_i^{\mathrm{AS}}\beta^{\mathrm{A}} + p_i^{\mathrm{US}}\beta^{\mathrm{U}}\right) \sum_j b_{ji}\epsilon_j \end{aligned} \tag{3.10}$$

and therefore

$$\sum_j \left[ \left(1 - (1-\gamma)p_i^{\mathrm{A}}\right) b_{ji} - \frac{\mu}{\beta^{\mathrm{U}}}\delta_{ji} \right] \epsilon_j = 0 \tag{3.11}$$

where $\delta_{ij}$ are the elements of the identity matrix. Note that the solution of Eq. 3.11 reduces to an eigenvalue problem for the matrix $H$ whose elements are $h_{ji} = (1-(1-\gamma)p_i^{\mathrm{A}})b_{ji}$. The onset of the epidemics is the minimum value of $\beta^{\mathrm{U}}$ satisfying Eq. 3.11. Denoting $\Lambda_{\max}(H)$ the largest eigenvalue of $H$, the critical point is written as

$$\beta_c^{\mathrm{U}} = \frac{\mu}{\Lambda_{\max}(H)}. \tag{3.12}$$

Note that $\beta_c$ depends explicitly on the dynamics on the virtual layer, in particular of the values of $p_i^{\mathrm{A}}$. Interestingly, if we consider the critical value $\lambda_c = \delta/\Lambda_{\max}(A)$ of the onset of awareness when decoupled from the infection, i.e. as a simple spreading process on the virtual layer with no interaction with the physical layer, then for $\lambda < \lambda_c$ Eq. 3.12 reduces to $\beta_c = \mu/\Lambda_{\max}(B)$, and the onset of the epidemics, is obviously independent of the awareness. The point $(\lambda_c, \beta_c)$ defines what we call a *meta-critical* point for the epidemic spreading. However, for values of $\lambda > \lambda_c$ the onset of the epidemics depends on the structure of the virtual layer and the dynamics of the awareness. Specifically, it depends on the stationary values of the probabilities $p_i^{\mathrm{A}}$ of the virtual layer, decoupled from the multiplex. These values are found by solving the fixed point equations of the virtual layer only.

The analytical results are crosschecked with extensive computer simulations of the coupled dynamics UAU-SIS in different configurations of multiplex. For the sake of simplicity, we will present the results for $\gamma = 0$, meaning that $\beta^{\mathrm{A}} = 0$ (and henceforth $q_i^A = 1$, and $\beta^{\mathrm{U}} = \beta$). This corresponds to complete immunity of nodes aware of the infection, although the calculation is identical for any other different value of $\gamma$. In Fig. 3.3 we plot the comparison of MMCA with Monte Carlo simulations, for a quenched multiplex of two layers, where in the physical layer we build a power-law degree distribution network generated with the configurational model with exponent 2.5 of 1000 nodes, and in the virtual layer we have the same network with 400 extra random links (non-overlapping with previous ones). We use this multiplex as a representation of a structure that could account for a realistic scenario, where all the connections in the physical layer are also present in the virtual network, and the virtual network has additional links. Nevertheless, different multiplex scenarios were explored, for the sake of completion, and in all of them the theory presented is equally accurate. Note that the MMCA approach is specially suited for quenched networks, and then it is not necessary to assess the validity of the approximation in the thermodynamic limit [90, 92]. The average accuracy of the approximation is about 2%.

We have also explored the full phase diagram $(\lambda - \beta)$ of the UAU-SIS dynamics for the same multiplex as before, see Fig. 3.4. We represent the fraction of infected individuals in the whole population, in the stationary state, $\rho^I$. The agreement is very good for the full phase space, being the relative error less than 2.5% in all the multiplex configurations explored, e.g. composing random homogeneous networks (Erdős-Rényi networks) and heterogeneous networks (scale-free networks), for different values of the parameters.

Finally, we are also able to plot the prediction of the critical epidemic threshold line $\beta_c(\lambda)$ given by Eq. 3.12 for different values of the recovery probabilities $\delta$ and $\mu$, see Fig. 3.5. Looking at the curves in Fig. 3.5 we observe that initially the epidemic threshold does not depend on the awareness. At a certain point $\lambda_c$, what we call the meta-critical point, the epidemics is delayed and contained. The region where the meta-critical point is localized corresponds to the area bounded by $[0, 1/\Lambda_{\max}(A)] \times [0, 1/\Lambda_{\max}(B)]$, see the shaded area of the figure.

Summarizing, the results show that the coexistence of different topologies spreading antagonistic effects raises interesting physical phenomena, for example the emergence of a meta-critical point, where the diffusion of awareness is able to control the onset of the epidemics. Given the specific nature of the aware-

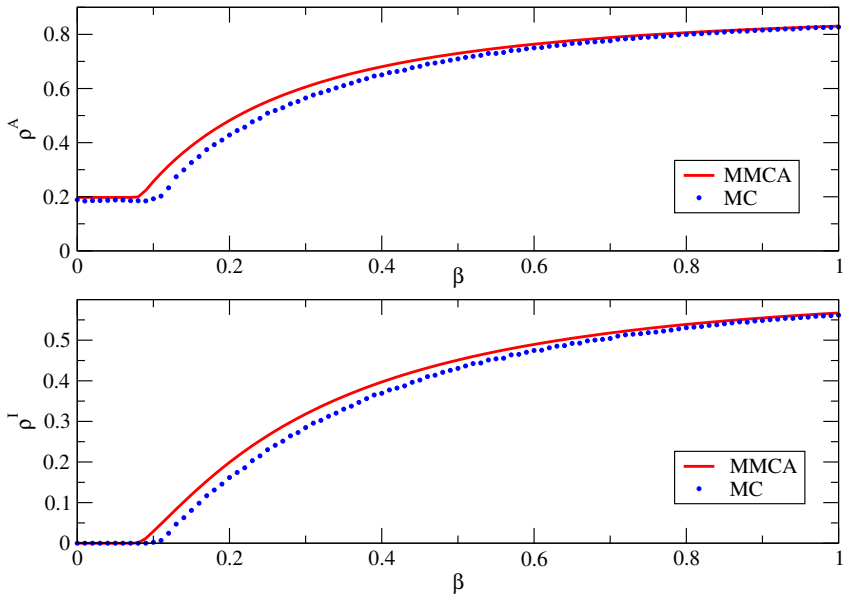



Figure 3.3: **Top.** Comparison of the stationary fraction of aware individuals $\rho^A = \frac{1}{N}\sum_i p_i^A$ using Monte Carlo (dotted line) simulations and the MMCA approach (solid line) as a function of the infectivity $\beta$ for a fixed value of $\lambda = 0.15$. **Bottom.** Comparison of the stationary fraction of infected individuals $\rho^I = \frac{1}{N}\sum_i p_i^I$ using Monte Carlo (dotted line) simulations and the MMCA approach (solid line) as a function of the infectivity $\beta$. The initial fraction of infected nodes is set to 0.2. The multiplex structure is, in this case: i) physical layer, a scale-free network of 1000 nodes generated with the configurational model, and with exponent 2.5, ii) virtual layer, the same network than in the physical layer but with 400 extra random links (non-overlapping with previous). The values for the recovery probabilities are $\delta = 0.6$, and $\mu = 0.4$.

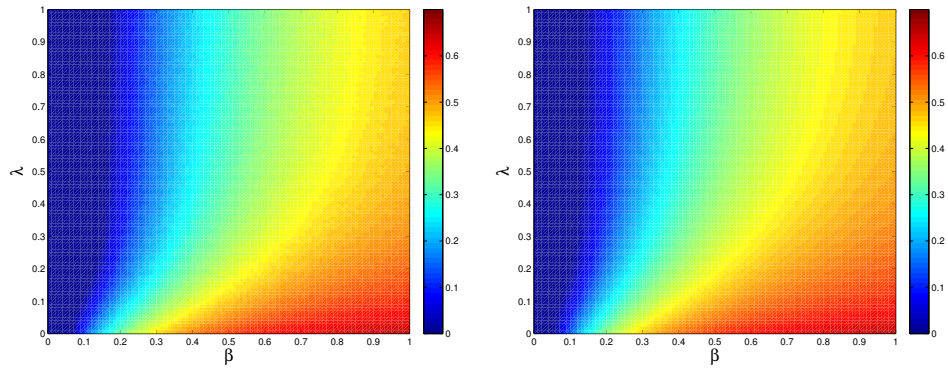



Figure 3.4: Comparison between Monte Carlo and MMCA for the fraction $\rho^I$ of infected individuals in the stationary state (colors represent the fraction of infected individuals). Left, full phase diagram $\lambda - \beta$ for the same multiplex described in Fig. 3.3 obtained by averaging 50 Monte Carlo simulations for each point in the grid $100 \times 100$. Right, same for the MMCA. The relative error for the full phase diagram is $\approx 1.6\%$.

ness spreading proposed here, equivalent to a SIS process, this setup is also valid to describe a generalist scenario of two competing infectious strains coexisting in a multiplex structure, for either the cases where strains reinforce or weaken each other. The genuine mechanism underlying the emergence of the dependence of the onset of the epidemics on the diffusion of the awareness is rooted to the cyclic character of both coupled processes. If one of the processes is not cyclic ($\delta = 0$ or $\mu = 0$) this dependence disappears. The high accuracy of the MMCA is specially useful in this scenario of coupled dynamics in quenched networks, where heterogeneous mean-field approximations for binary states, or in general approximations for annealed networks [170, 154, 86] could be difficult to define because of the structure of the multiplex, where the degree-class is multivalued. The results provide clues to quantify the effect of the word of mouth — transmitted for example via Facebook, or Twitter— in campaigns against seasonal diseases, and sheds light on the power of information to decrease the incidence of epidemics or even its eradication.

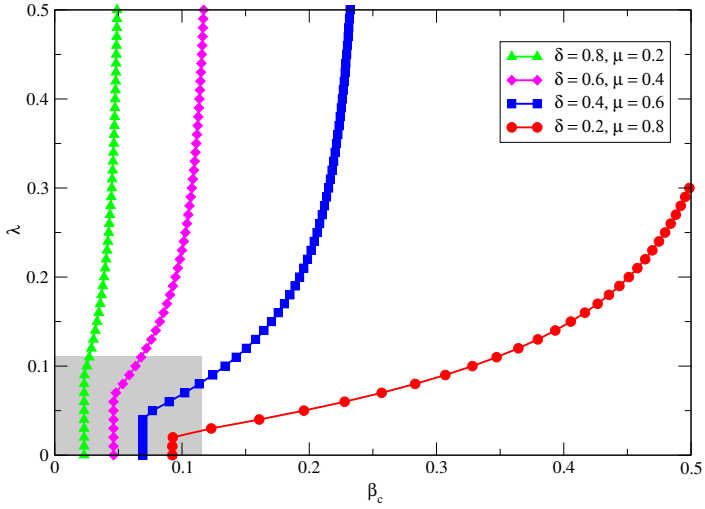



Figure 3.5: Dependence of the onset of the epidemics $\beta_c$ as a function of $\lambda$ computed using Eq. 3.12, for different values of the recovery $\delta$ and $\mu$, for the same multiplex described in Fig. 3.3. The shaded rectangle corresponds to the area where the meta-critical points may be, which are bounded by the topological characteristics of the multiplex $1/\Lambda_{\max}(A)$ and $1/\Lambda_{\max}(B)$.

## 3.2 EPIDEMIC SPREADING IN THE PRESENCE OF LOCAL AND GLOBAL AWARENESS

In the previous section we presented a setup for interacting epidemics and information spreading, on top of multiplex networks. It consisted on a two layer structure, where each spreading process diffused in a different layer. We studied the interaction between the two spreading processes and found an interesting metacritical point in the phase space of the epidemics. Two of the hypotheses used in the previous model were: (I) getting infected with the disease supposed an immediate change to the *aware* state, and (II) being aware of the disease implied total immunization. Even though these two suppositions are plausible, they are too restrictive if we wish to understand more realistic situations. Therefore, here we study a more general scenario, where we relax the *self-awareness* and *immunization* conditions and study the consequences of these two effects on the onset of the epidemics and the final fraction of infected nodes.

Additionally, we also wish to take into account an important agent in information diffusion: the *mass media*. In this generalized model we include the effect of massive awareness information flowing through the network, in order to assess whether an external node with massive connectivity is crucial or not to the final outcome of the epidemics.

As we will see, the study suggests that the degree of immunization is able to shift the epidemic threshold, while the effects of the self-awareness are residual. Moreover, we find that the presence of the mass media makes the metacritical point of the epidemics vanish.

### 3.2.1 MODEL FOR AWARENESS AND EPIDEMIC SPREADING WITH MASS MEDIA

The generalized UAU-SIS model presented here accounts for the effect that awareness has on the spreading of a disease in the scenario where awareness does not imply total immunization and being infected does not suppose an immediate spreading of information. The relaxation of these two conditions account for the possibility that nodes do not have full access to the mechanisms designed to prevent infection, and for nodes that decide not to actively disseminate information of the disease in their contacts network, respectively. This latter case could also

refer to those situations in which the infected patient does not know that he is infected until some time has passed, or for people who simply do not wish to declare their health state to other people.

To mathematically include the previous conditions in the existing UAU-SIS framework, we make use of the probabilities $\kappa$ and $\gamma$ accounting for the probability of self-awareness and immunization, respectively. The UAU and SIS processes will have a spreading probability of $\lambda$ and $\beta$, and recovery probabilities of $\delta$ and $\mu$, respectively. The interaction between both processes is modeled as follows: a node that is infected in the SIS layer will become aware in the UAU layer with probability $\kappa$. Similarly, a node that is aware on the UAU layer will take measures for preventing infection, therefore the parameter $\gamma$ regulates the probability of a node to get infected. The infectivity parameter of the SIS is then different depending on the state of the node in the information layer. The term $\beta^{\mathrm{U}}$ regulates the probability of a node to get infected when it is unaware of the disease, while $\beta^{\mathrm{A}} = \gamma\beta^{\mathrm{U}}$ regulates the probability when the node is aware. Thus, the parameter $\gamma$ ranges from 0, representing total immunization, to 1, representing no effect of the information awareness on the epidemics. Equivalently, we can regulate the upwards interaction by tuning the parameter $\kappa$ from 0 to 1. Note that when $\gamma = 1$ and $\kappa = 0$ the two interactions are disabled and the setup becomes equivalent to running both processes in single-layer independent networks. Oppositely, when $\gamma = 0$ and $\kappa = 1$, the system is maximally coupled and we recover the scenario presented in the previous section.

Additionally, we wish to take into account the effect that mass-media entities may have on the spreading. Mass media are entities with eventually a large impact (connectivity) that regularly transmit information over the entire population. Moreover, they are often perceived as reliable sources of information, and therefore its role when warning from an epidemics outbreak may be crucial to the final outcome of the disease. To incorporate this effect to our setup, we add a single node connected to all nodes in the UAU layer. This node participates in the UAU dynamics, however its state is permanently aware, and it is constantly contacting nodes from the UAU layer to convert them to the aware state at with probability $m$. This behavior is then equivalent to: at each time step, all nodes have the chance to become aware through the UAU dynamics, but if they are not converted by their neighbors, they may spontaneously become aware with probability $m$.

In Fig. 3.6 we depict a sketch of the resulting scenario. Note that in this extended model, the $N$ nodes can be in one of the four following states: US (Unaware and Susceptible), UI (Unaware and Infected), AS (Aware and Susceptible) or AI (Aware and Infected), which correspond to the same three states of the previous UAU-SIS model but here the UI state is allowed. Proceeding in the same way as in the previous scenario, we might construct the four transition probability trees (see Figs. 3.7 and 3.8), which now account for the transition induced by the mass media.

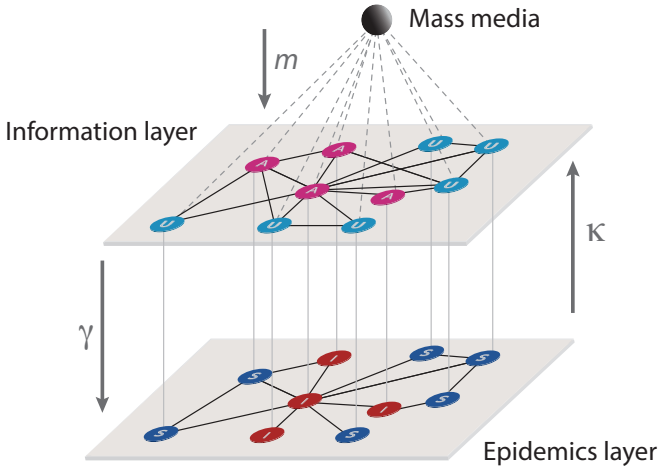



Figure 3.6: Awareness-epidemic model in the presence of mass media. The upper (information) layer is supporting the spreading of awareness, and nodes have two possible states: unaware (U) or aware (A). The lower (epidemic) layer corresponds to the network where the epidemic spreading takes place. The nodes are the same actors than in the upper layer, but here their state can be: susceptible (S) or infected (I). The mass media is represented as a top node that provides with information to the full system.

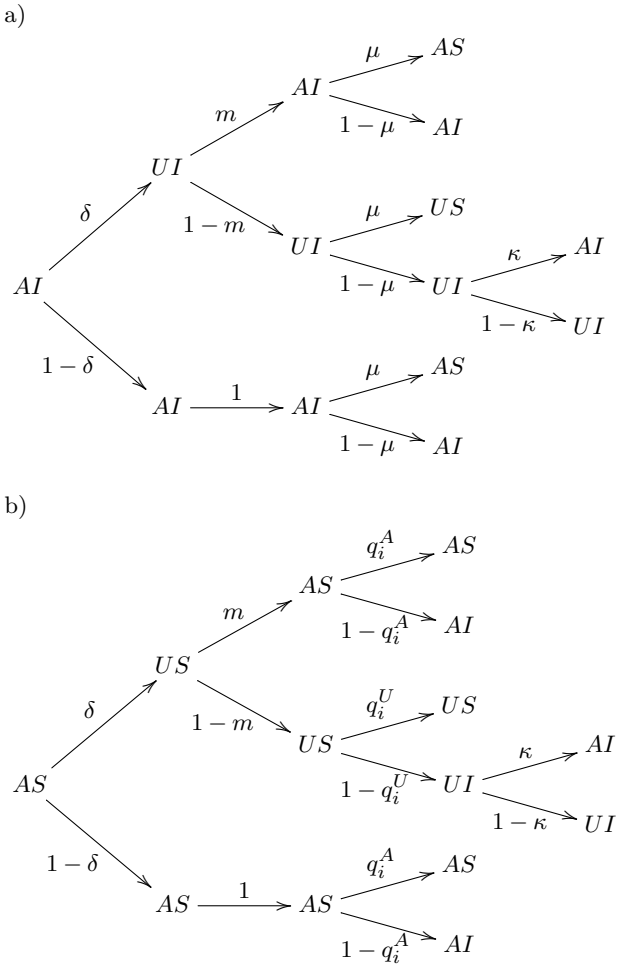



Figure 3.7: Transition probability trees for the AI (a) and AS (b) states. The root of each tree represents the state of any node at time $t$, and the leaves their states at time $t + 1$. Each time step is subdivided in four phases: awareness spreading (UAU process), mass media broadcast, epidemic spreading (SIS process) and self-awareness of being infected.

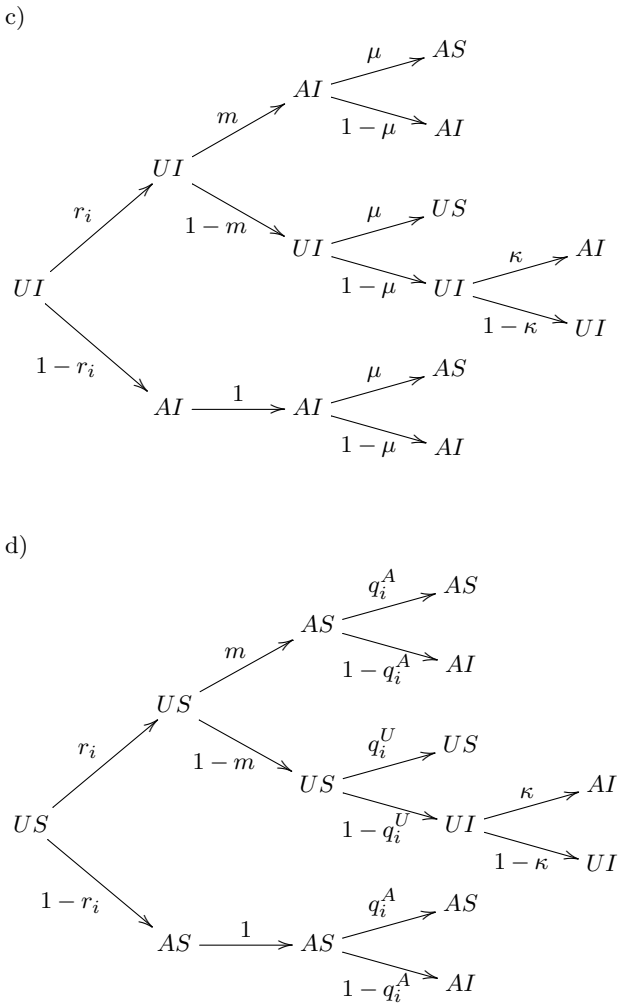



Figure 3.8: Transition probability trees for the UI (c) and US (d) states.

The MMCA equations of these models are calculated following the same procedure as in the previous model. Starting out from the transition probability trees we calculate the four equations, which read:

$$\begin{aligned} p_i^{\mathrm{US}}(t+1) &= p_i^{\mathrm{UI}}(t) r_i(t)(1-m)\mu + p_i^{\mathrm{AI}}(t)\delta(1-m)\mu \\ &+ p_i^{\mathrm{US}}(t) r_i(t)(1-m) q_i^{\mathrm{U}}(t) + p_i^{\mathrm{AS}}(t)\delta(1-m) q_i^{\mathrm{U}}(t)\,, \end{aligned} \tag{3.13}$$

$$\begin{aligned} p_i^{\mathrm{UI}}(t+1) &= p_i^{\mathrm{UI}}(t) r_i(t)(1-m)(1-\mu)(1-\kappa) + p_i^{\mathrm{AI}}(t)\delta(1-m)(1-\mu)(1-\kappa) \\ &+ p_i^{\mathrm{US}}(t) r_i(t)(1-m)(1-q_i^{\mathrm{U}}(t))(1-\kappa) \\ &+ p_i^{\mathrm{AS}}(t)\delta(1-m)(1-q_i^{\mathrm{U}}(t))(1-\kappa)\,, \end{aligned} \tag{3.14}$$

$$\begin{aligned} p_i^{\mathrm{AS}}(t+1) &= p_i^{\mathrm{UI}}(t)[r_i(t)m\mu + (1-r_i(t))\mu] + p_i^{\mathrm{AI}}(t)[\delta m\mu + (1-\delta)\mu] \\ &+ p_i^{\mathrm{US}}(t)[r_i(t)m q_i^{\mathrm{A}}(t) + (1-r_i(t)) q_i^{\mathrm{A}}(t)] \\ &+ p_i^{\mathrm{AS}}(t)[\delta m q_i^{\mathrm{A}}(t) + (1-\delta) q_i^{\mathrm{A}}(t)]\,, \end{aligned} \tag{3.15}$$

$$\begin{aligned} p_i^{\mathrm{AI}}(t+1) &= p_i^{\mathrm{UI}}(t)[r_i(t)m(1-\mu) + r_i(t)(1-m)(1-\mu)\kappa + (1-r_i(t))(1-\mu)] \\ &+ p_i^{\mathrm{AI}}(t)[\delta m(1-\mu) + \delta(1-m)(1-\mu)\kappa + (1-\delta)(1-\mu)] \\ &+ p_i^{\mathrm{US}}(t)[r_i(t)m(1-q_i^{\mathrm{A}}(t)) + r_i(t)(1-m)(1-q_i^{\mathrm{U}}(t))\kappa + (1-r_i(t))(1-q_i^{\mathrm{A}}(t))] \\ &+ p_i^{\mathrm{AS}}(t)[\delta m(1-q_i^{\mathrm{A}}(t)) + \delta(1-m)(1-q_i^{\mathrm{U}}(t))\kappa + (1-\delta)(1-q_i^{\mathrm{A}}(t))]\,, \end{aligned} \tag{3.16}$$

where $r_i(t)$, $q_i^{\mathrm{A}}(t)$ and $q_i^{\mathrm{U}}(t)$ are defined as in Eqs. 3.4-3.6.

### 3.2.2 DETERMINING THE ONSET OF THE EPIDEMICS IN PRESENCE OF GLOBAL AWARENESS

Starting out from the MMCA equations derived from the transition probability trees, one can calculate the critical epidemic threshold $\beta_c^{\mathrm{U}}$ as a function of the rest of the parameters in the system at the stationary state $p_i(t+1) = p_i(t)$ for all nodes $i$ and all states. First, since this epidemic threshold is given by the order parameter $\rho^{\mathrm{I}}$, which corresponds to the fraction of infected nodes in the system and is calculated as

$$\rho^{\mathrm{I}} = \frac{1}{N}\sum_{i=1}^{N} p_i^{\mathrm{I}} = \frac{1}{N}\sum_{i=1}^{N}(p_i^{\mathrm{UI}} + p_i^{\mathrm{AI}})\,, \tag{3.17}$$

it is useful to add Eqs. 3.14 and 3.16 to obtain, in the steady state,

$$\begin{aligned} p_i^{\mathrm{I}} &= p_i^{\mathrm{I}}(1-\mu) \\ &+ p_i^{\mathrm{US}}[r_i(1-m)(1-q_i^{\mathrm{U}}) + (1-r_i(1-m))(1-q_i^{\mathrm{A}})] \\ &+ p_i^{\mathrm{AS}}[\delta(1-m)(1-q_i^{\mathrm{U}}) + (1-\delta(1-m))(1-q_i^{\mathrm{A}})]\,. \end{aligned} \tag{3.18}$$

Near the onset of the epidemics, the probability of nodes to be infected is close to zero, i.e. $p_i^{\mathrm{I}} = \epsilon_i \ll 1$. Accordingly, Eqs. 3.6 and 3.5 are approximated as:

$$q_i^{\mathrm{U}} \approx 1 - \beta^{\mathrm{U}} \sum_j b_{ji}\epsilon_j = 1 - \sigma_i\,, \tag{3.19}$$

$$q_i^{\mathrm{A}} \approx 1 - \gamma\beta^{\mathrm{U}} \sum_j b_{ji}\epsilon_j = 1 - \gamma\sigma_i\,, \tag{3.20}$$

where

$$\sigma_i = \beta^{\mathrm{U}} \sum_j b_{ji}\epsilon_j\,, \tag{3.21}$$

and Eq. 3.18 becomes

$$\begin{aligned} \epsilon_i &= \epsilon_i(1-\mu) \\ &\quad + p_i^{\mathrm{US}}[r_i(1-m)\sigma_i + (1-r_i(1-m))\gamma\sigma_i] \\ &\quad + p_i^{\mathrm{AS}}[\delta(1-m)\sigma_i + (1-\delta(1-m))\gamma\sigma_i] \\ &= \epsilon_i(1-\mu) \\ &\quad + [p_i^{\mathrm{U}} r_i(1-m) + p_i^{\mathrm{A}}\delta(1-m)]\sigma_i \\ &\quad + [p_i^{\mathrm{U}}(1-r_i(1-m)) + p_i^{\mathrm{A}}(1-\delta(1-m))]\gamma\sigma_i\,. \end{aligned} \tag{3.22}$$

Here we have made use of $p_i^{\mathrm{U}} = p_i^{\mathrm{US}} + p_i^{\mathrm{UI}} \approx p_i^{\mathrm{US}}$ and $p_i^{\mathrm{A}} = p_i^{\mathrm{AS}} + p_i^{\mathrm{AI}} \approx p_i^{\mathrm{AS}}$, since $\epsilon_i = p_i^{\mathrm{UI}} + p_i^{\mathrm{AI}} \ll 1$. In a similar way, removing $O(\epsilon_i)$ terms in the stationary state of Eqs. 3.13 and 3.15 we get

$$p_i^{\mathrm{U}} = p_i^{\mathrm{U}} r_i(1-m) + p_i^{\mathrm{A}}\delta(1-m)\,, \tag{3.23}$$

$$p_i^{\mathrm{A}} = p_i^{\mathrm{U}}(1-r_i(1-m)) + p_i^{\mathrm{A}}(1-\delta(1-m))\,. \tag{3.24}$$

These last equations correspond to an UAU process with mass media decoupled from the SIS process, with $p_i^{\mathrm{U}} + p_i^{\mathrm{A}} = 1$. Substituting them in Eq. 3.22 leads to

$$\begin{aligned} \epsilon_i &= (1-\mu)\epsilon_i + p_i^{\mathrm{U}}\sigma_i + p_i^{\mathrm{A}}\gamma\sigma_i \\ &= (1-\mu)\epsilon_i + (p_i^{\mathrm{U}} + p_i^{\mathrm{A}}\gamma)\beta^{\mathrm{U}} \sum_j b_{ji}\epsilon_j \,, \end{aligned} \tag{3.25}$$

which can be written as:

$$\sum_j \left[\beta^{\mathrm{U}}(p_i^{\mathrm{U}} + \gamma p_i^{\mathrm{A}})b_{ji} - \mu\delta_{ij}\right]\epsilon_j = 0\,, \tag{3.26}$$

where $\delta_{ij}$ are the elements of the identity matrix. Defining matrix $H$ with elements

$$h_{ij} = (p_i^{\mathrm{U}} + \gamma p_i^{\mathrm{A}})b_{ji}\,, \tag{3.27}$$

the non-trivial solutions of Eq. 3.26 are eigenvectors of $H$, whose largest real eigenvalues are equal to $\mu/\beta^{\mathrm{U}}$. Therefore, the onset of the epidemics is given by the largest real eigenvalue of $H$,

$$\beta_c^{\mathrm{U}} = \frac{\mu}{\Lambda_{\max}(H)}\,. \tag{3.28}$$

Note that matrix $H$ depends on the solutions of Eqs. 3.23 and 3.24, or equivalently

$$p_i^{\mathrm{A}} = (1-p_i^{\mathrm{A}})(1-r_i(1-m)) + p_i^{\mathrm{A}}(1-\delta(1-m))\,, \tag{3.29}$$

where

$$r_i = \prod_j \left(1 - a_{ji}p_j^{\mathrm{A}}\lambda\right), \tag{3.30}$$

which are also solved by iteration.

Now we are in the position of investigating the effects of the three main parameters of the model: $\gamma$ (degree of immunization), $\kappa$ (self-awareness) and $m$ (mass media). For the sake of consistency, the multiplex setup used for all the following tests is the same as in the previous model, which is described in Fig. 3.3. In the following, we analyze the incidence of epidemics for different values of the parameters. To simplify the notation we will denote $\beta^U$ by $\beta$.

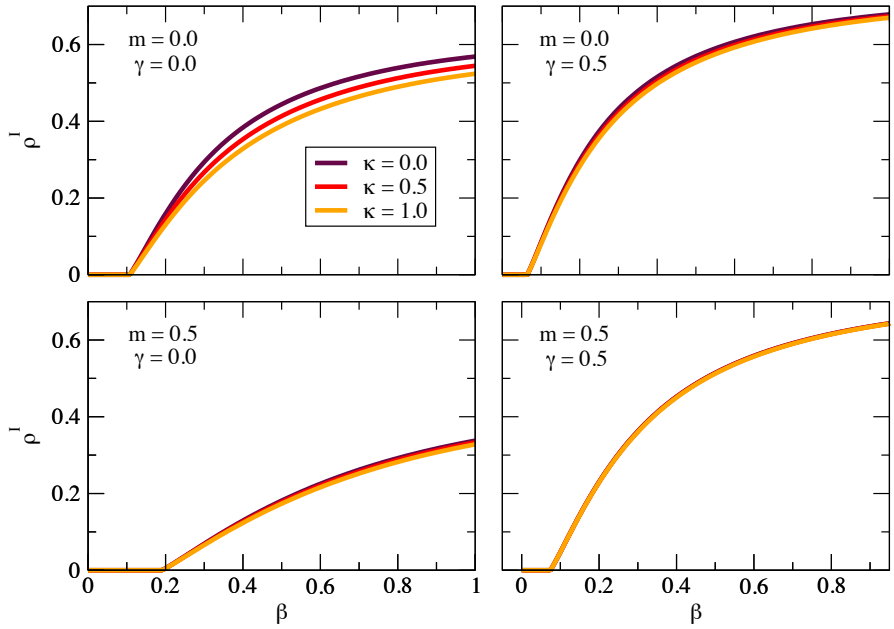



Figure 3.9: Fraction of infected nodes as a function of the infectivity parameter $\beta$, for different values of the self-awareness parameter $\kappa$. The networks used in this setup are the same as throughout the document. The rest of the values of parameters are: $\lambda = 0.3$, $\delta = 0.6$, $\mu = 0.4$. The panel shows: top-left corner $\gamma = 0.0$ (i.e. total immunization) and the mass media effect is turned off( $m = 0$); top-right corner the immunization is reduced at $\gamma = 0.5$; bottom-left corner mass media effect active ($m = 0.5$) with total immunization ($\gamma = 0.0$); and bottom-right corner mass media active ($m = 0.5$) and partial immunization ($\gamma = 0.5$).

In Fig. 3.9 we plot the fraction of infected nodes as a function of $\beta$ for different values of the self-awareness parameter $\kappa$. The rest of the parameters are set to intermediate values (see caption). The $\kappa$ parameter regulates the probability to be aware of your own disease, that is, the probability of going from UI state to AI state. In the figure panel we consider that the mass media is inactive (top) or active (bottom), and also that the immunization is total (left) or partial (right). Observing the figures, we see that by varying $\kappa$ the onset of the epidemics is not affected in any of the scenarios, which is a consequence of the absence of $\kappa$ in the Eqs. 3.27 to 3.30 to determine the epidemic threshold. We also observe no significant change on the final incidence. This result indicates that the incidence of the self-awareness in the whole process is negligible, even in the limiting cases

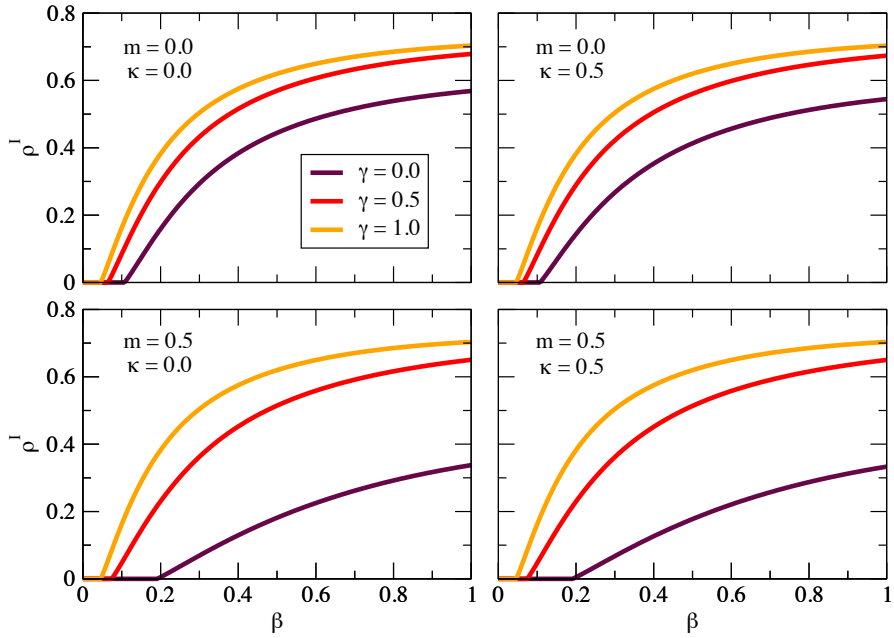



Figure 3.10: Fraction of infected nodes as a function of the infectivity parameter $\beta$, for different values of the immunization parameter $\gamma$. The rest of the values of parameters are set to $\lambda = 0.3$, $\delta = 0.6$ and $\mu = 0.4$. The top-left panel shows the results for inexistent mass media and self-awareness; top-right has an intermediate self-awareness $\kappa = 0.5$ keeping mass media turned off; bottom-left has no self-awareness but an intermediate mass media effect $m = 0.5$ and bottom-right is set to intermediate values of both $m$ and $\kappa$.

where infected unaware individuals remain unaware of its sickness ($\kappa = 0$) or certainly become aware of it ($\kappa = 1$), thus concluding that the self-awareness is not a key factor for the dynamical behavior of our system.

In Fig. 3.10 we plot the density of infected nodes for different values of $\gamma$ for four combinations of the parameters $m$ and $\kappa$. The panel depicts the scenarios where the mass media is inactive (top) or active (bottom), and also that the self awareness is non-existent (left) or existent (right). The parameter $\gamma$ accounts for the immunity that a node gains when it is aware of the disease. We observe that for low values of $\gamma$ (high immunity) the final incidence of the epidemics is lowered, and the critical point is shifted right (comparing with non-existent coupling $\gamma = 1$), for whatever the values of $m$ and $\kappa$. We can also see, if we

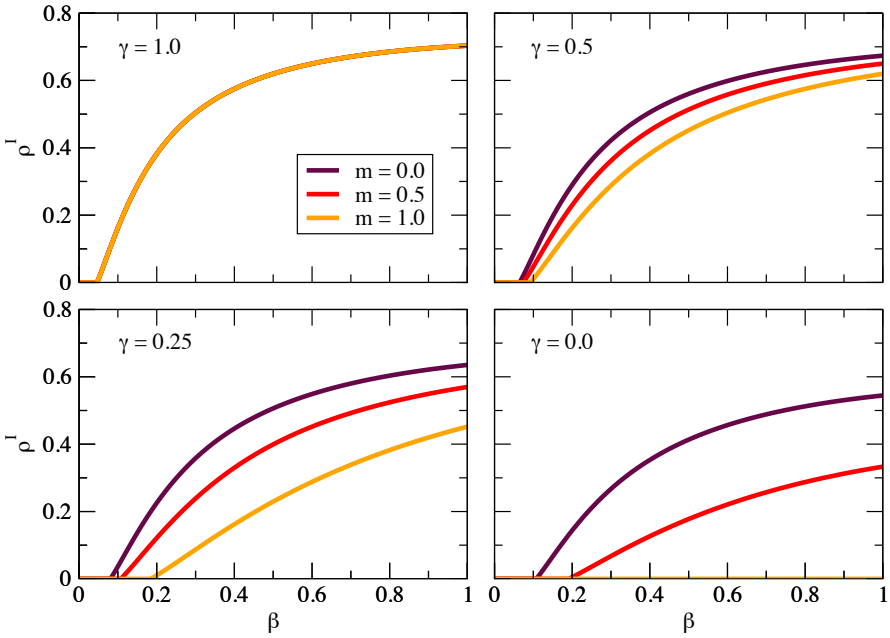



Figure 3.11: Fraction of infected nodes as a function of the infectivity parameter $\beta$, for different values of the parameter representing the mass media effect, $m$. The networks used in this setup are the same as throughout the document. The rest of the values of parameters are: $\lambda = 0.3$, $\delta = 0.6$, $\mu = 0.4$ and $\kappa$ is fixed to 0.5. The four panels correspond to values of $\gamma = 1$, $\gamma = 0.50$, $\gamma = 0.25$ and $\gamma = 0$.

compare the left and right plots, that $\kappa$ does not change the onset of the epidemics, and only slightly the final incidence, as explained previously.

To analyze the effect of the mass media on this model we also plot the density of infected nodes for different values of $m$, see Fig. 3.11. We fix the value of $\kappa$ to 0.5 and move only the immunization parameter $\gamma$. From this picture we observe that high values of $m$ shift the onset of the epidemics right and lower the final incidence. For low values of $\gamma$ the mass media effect is very pronounced (see bottom-right panel); on the other hand, when $\gamma = 1$ (top-left plot) the mass media has no effect whatsoever as the epidemic layer has effectively been disconnected from the information layer.

We have shown how the critical point is affected by the parameters $\gamma$ and $m$. In Fig. 3.12 we plot the non-linear dependence of $\beta_c$ on the degree of immunization

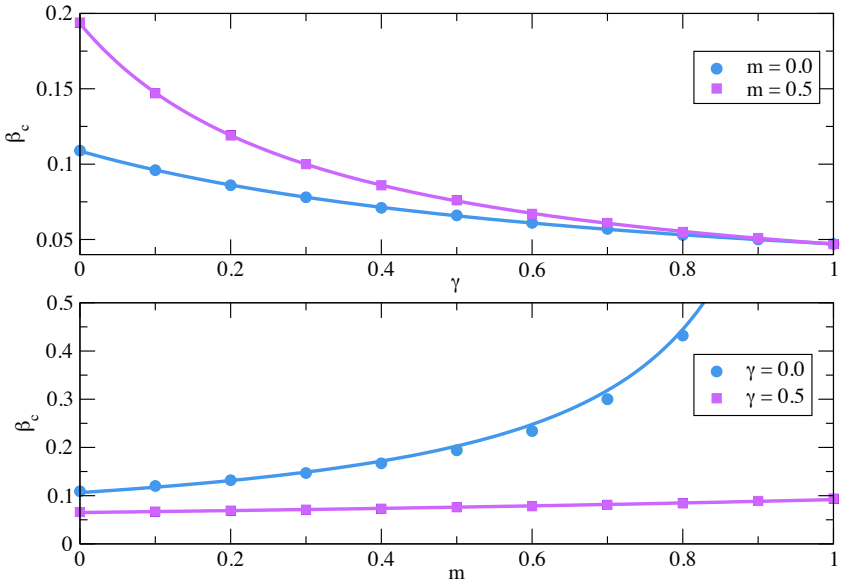



Figure 3.12: Plot of $\beta_c$ as a function of $\gamma$ (top) and $m$ (bottom). Dots represent the data and the line is the fitting function $\beta_c \sim (a + bx)^{-1}$. As described, the less intense the immunization degree (larger $\gamma$) the lower the critical point of the epidemics, and conversely, the larger the intensity of the mass media $m$ the larger the critical onset.

and the mass media. Surprisingly enough, in both cases the dependence can be empirically fitted to an expression that is inversely linear to the parameters, i.e. $\beta_c \sim (a + bx)^{-1}$ for certain constants $a$ and $b$ and being the variable $x = \gamma$ or $x = m$.

The most remarkable outcome of the mass media effect is observed when representing the curve of critical points in the $\lambda - \beta$ phase space. As seen in the previous work, the coupling of the UAU and SIS layer implies the existence of a metacritical point, a point from which on the critical onset of the epidemics depends on the incidence of awareness in the population. The effect of the mass media is that of making the metacritical point vanish, which can be observed in Fig. 3.13. When $m = 0$ we see that for low values of the awareness infectivity parameter $\lambda$, the onset of the epidemics $\beta_c$ is independent of $\lambda$, and it is not until a certain point (the metacritical point) that the UAU process begins to influence the onset of the epidemics. However, when the mass media is greater than zero

(even for very small values) the phenomenology is different and the metacritical point disappears. The explanation of this phenomenon is rooted on the nature of the awareness provided by the mass media. Being it a random process acting on the whole population at each time step, whatever the capability of the awareness to spread it, a certain finite fraction of aware individuals will survive. This pool of aware individuals effectively decrease the epidemics, although not in a uniform linearly predictable way.

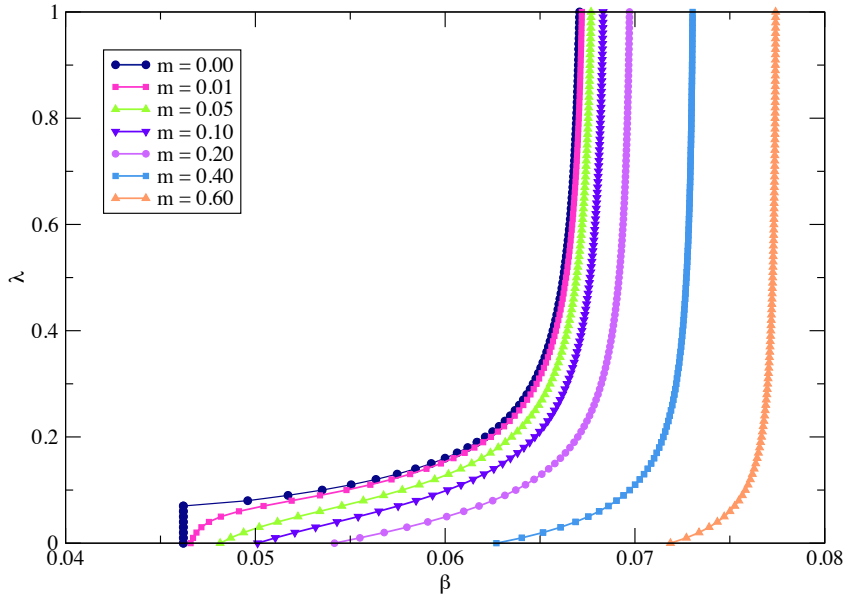



Figure 3.13: Value of the onset of the epidemics as a function of the UAU parameter $\lambda$, for different values of the mass media parameter $m$. The setup is the same used throughout this document, with $\delta = 0.6$, $\mu = 0.4$ and parameters $\kappa = 1.0$ and $\gamma = 0.0$ which imply maximum coupling between layers.

Summarizing, we have studied the effects of the interplay between awareness and disease, when both spreading processes take place on the same nodes, but on two different connectivity layers. We have done this by means of a model which assumes that infected individuals get immediately aware, that aware individuals get immediately immunized, and that no awareness massive broadcast is present (mass media). The main physical result on such system relies on the emergence of a metacritical point, where the critical onsets of both dynamics get intertwined and the onset of the epidemics starts depending on the incidence of aware in-

dividuals. In the generalized model, we relax the two first assumptions, and included the presence of a mass media. We have found analytic expressions using a Microscopic Markov Chain Approach, that relate the decrease of the epidemic incidence with the increase in the level of immunization, and the modification of the incidence due to the mass media. The non-linear character of the relation between the two processes makes the analytical approach extremely useful to understand different scenarios. The results obtained are interesting: while the immediacy of awareness when infected (self-awareness) has almost no effect on the dynamics, the other two factors, namely the degree of immunization of aware individuals and the mass media, do change the critical aspects of the epidemics spreading. Most remarkably, the presence of a highly active mass media agent is able to make the meta-critical point vanish.

# 4

# ON THE ANALYSIS OF NEUROSCIENCE DATA

The theory of complex networks has proven to be a useful framework for the study of the interplay between structure and functionality in social, technological, and biological systems. The analysis of such resulting abstraction of the system, the network, provides clues about regularities that can be connected with certain functionalities, or even be related to organization mechanisms that help to understand the rules behind the system's complexity. Particularly, in biological systems, the characterization of the emergent self-organization of their components is of utmost importance to comprehend the mechanisms of life [65, 205, 98].

One of the major challenges in biology and neuroscience is the ultimate understanding of the structure and function of neuronal systems, in particular the human brain, whose representation in terms of complex networks is especially appealing [36, 199]. Network theory and its mathematical framework are able to provide statistical measures that highlight key topological features of the networks under study. Applied to neuroscience, these measures have facilitated the comprehension of processes as complex as brain development [200], learning [17] and dysfunction [190, 222]. Particularly, these measures have unfolded new relationships between brain dynamics and functionality. For instance, synchronization between neuronal assemblies in the developing hippocampus has been ascribed to the existence of super-connected nodes in a scale-free topology [30]; efficient information transfer has been associated to circuits with small-world features [129], such as in the the nematode worm *C. elegans* [218] or the brain cortex [201]; and the coexistence of both segregated and integrated activity in the brain has been hypothesized to arise from a modular circuit architecture [101, 143].

An accessible approach to the understanding of neuronal systems are *in vitro* neuronal cultures. Indeed, neurons from the cortex or other brain areas can be

dissociated, seeded over a substrate, and cultured along several weeks. After, neurons self-organize, connecting to one another to create a *de novo* neuronal network with rich spontaneous activity patterns. If the mobility of those neurons is not intentionally impaired, they self-organize into well defined clusters which internally reach a coherent state of collective firing. During the maturation process of the culture, bundles of axons are formed between clusters. The analysis of the resulting structure of such cultures is still a challenge to the scientific community because it is extremely difficult to discern the links that carry electrical activity from those who do not, which makes the real physical connectivity structure inaccessible to observation. In the following, I present a collaborative, experimental work focused on characterizing the functional structure of clustered neuronal cultures, aimed to shed light on the interrelation between structure and functionality of networks of neurons.

## 4.1 CHARACTERIZING THE FUNCTIONAL STRUCTURE OF CLUSTERED CULTURED NEURONS

The analysis of the activity of neuronal cultures is considered to be a good proxy of the functional connectivity of *in vivo* neuronal tissues. Thus, the functional complex network inferred from activity patterns is a promising way to unravel the interplay between structure and functionality of neuronal systems. Here, we monitor the spontaneous self-sustained dynamics in neuronal clusters formed by interconnected aggregates of neurons. The analysis of the time delays between ignition sequences of the clusters allows the reconstruction of the *directed* functional connectivity of the network. We propose a method to statistically infer this connectivity and analyze the resulting properties of the associated complex networks. Surprisingly enough, in contrast to what has been reported for many biological networks, the clustered neuronal cultures present assortative mixing connectivity values, meaning that there is a preference for clusters to link to other clusters that share similar functional connectivity. These results suggest that the grouping of neurons and the assortative connectivity between clusters are intrinsic survival mechanisms of the culture.

### 4.1.1 EXPERIMENTAL DATA

For this work, we used experimental data curated and provided by scientists Sara Teller and Jordi Soriano at the Universitat of Barcelona. In the following, the experimental setup and signal treatments are briefly explained. For a more in-detail explanation of the whole procedure, see Appendix A.1.

The experiments were done using rat cortical neurons. Neurons were dissociated and seeded homogeneously on a glass substrate, limited to circular areas of 3mm in diameter for better control and full monitoring capability. The lack of adhesive proteins in the substrate rapidly favored cell–to–cell attachment and aggregation, giving rise to clustered cultures that evolved quickly. By *day in vitro* (DIV) 2, cultures already contained dozens of small aggregates, which coalesced and grew in size as the culture matured. Connections between clusters as well as initial traces of spontaneous activity were observed as early as DIV 5. At this stage cultures comprised of $20-30$ interconnected clusters, and were sufficiently stable and rich in activity for measurements. Although the strength of the connections in the network and its dynamics evolved further, we observed that the size and position of the clusters remained stable. Therefore, the dynamics were measured already at DIV 5, and studied cultures up to DIV 16. Fig. 4.1 (a) shows an image of a culture at DIV 14. Clusters appear as circular objects with an average diameter of 150 $\mu$m and a typical separation of 250 $\mu$m. Connections between clusters are visible as straight filaments that contain several axons.

Spontaneous activity of the clustered network was monitored through fluorescence calcium imaging, see Fig. 4.1 (b). Fluorescence images of the clustered network were acquired at a rate of 83–100 frames per second, and with an image size and resolution that allowed the monitoring of all the clusters in the network with sufficient image quality. Activity was recorded for typically 1 hour, which provided sufficient statistics in firing events while minimizing culture degradation due to photo-damage. The analysis of the images at the end of the measurement provided the variations in fluorescence intensity for each cluster and the corresponding onset times of firing.

As shown in Fig. 4.2, the average fluorescence signal of the network is characterized by peaks of intense cluster activity combined with silent intervals. The accompanying raster plot reveals that this activity actually corresponds to the collective ignition of a small group of clusters, which fire sequentially in a short

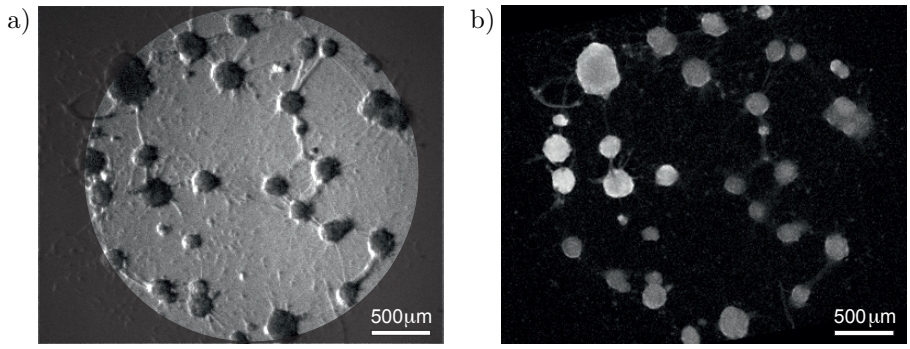



Figure 4.1: **a)** Bright field image of a network at *day in vitro* 14. Dark circular objects are aggregates of neurons (clusters), and filaments are visible physical connections between them. **b)** Corresponding fluorescence image, integrated over 50 frames ($\approx$ 0.5 s). Bright clusters at the top-left corner are active ones.

time window on the order of few hundred milliseconds. We denote by *bursts* these fast sequences of clusters activation.

We observed that the time spanned between two consecutively firing clusters typically ranged between 10 and 100 ms, as also observed by others [208, 220]. These times are fairly large compared to the eventual scale of signal integration–propagation between single neurons ($\approx$ 5ms), and is related to the large time scales associated to integration of the intra–clusters information. No consecutive activations were observed above 200 ms, signaling the termination of a burst. We therefore use this value of 200 ms as a cut–off to separate a given burst from the preceding one. Then, we hypothesize that two clusters that fired above 200 ms cannot be influenced by one another and therefore are not causally connected.

Bursts occurred every 30 seconds on average for the experiment shown in Fig. 4.2 and, as illustrated by the yellow bands in this figure, each burst typically encompasses a subset of clusters rather than the entire network. In general, however, the number of participating clusters within a burst depended on the details of the culture. Although in a typical experiment the collective firing comprised between 2 and 10 clusters, in some experiments the entire cluster population lighted up in a single bursting episode.

The analysis of the onset times of firing provides the cluster's activation sequence within each burst. As an example, Fig. 4.3 (a) depicts a highly active region of the network shown in Fig. 4.1 (b). This region contains 13 clusters,

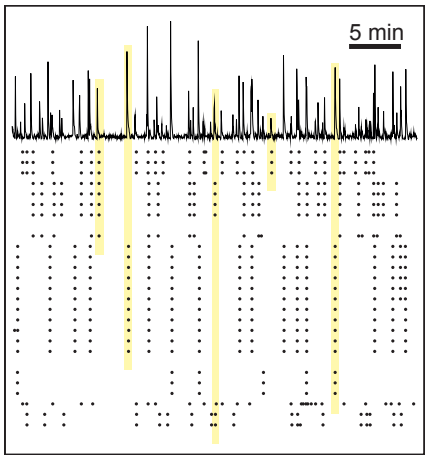



Figure 4.2: Spontaneous activity in the network. The top plot shows the average fluorescence signal of the clustered network shown in Fig. 4.1 (b), along 40 min of recording. The sharp peaks in fluorescence correspond to the fast sequential ignition of a group of clusters (burst). The bottom raster plot shows the clusters that ignite along the recording. The yellow bars relate a fluorescence peak with the ignition of a group of clusters, where each row represents a different cluster, and highlights the tendency for the clusters to activate in specific groups.

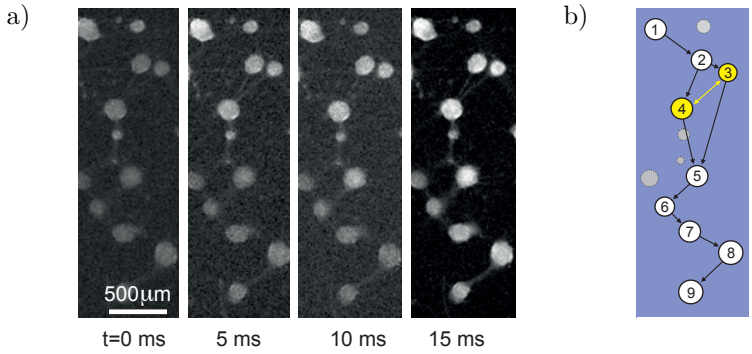



Figure 4.3: **a)** Example of a particular ignition sequence in a region of the network containing 13 clusters. From left to right, the progress of cluster's activation is revealed by the increase in fluorescence signal of the downstream connected clusters. **b)** Order of activation (black arrows) according to the analysis of the fluorescence signal. The clusters marked in yellow are those that fire simultaneously within experimental resolution. Grey nodes are clusters that do not participate in the firing sequence, and either fire independently or remain silent.

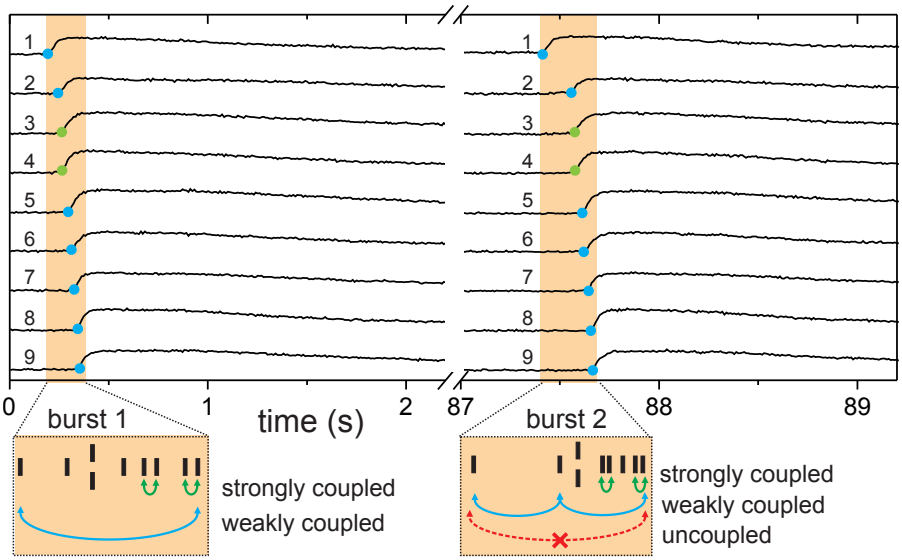



Figure 4.4: Detail of the fluorescence traces for the 9 participating clusters of Fig. 4.3 (b) along two consecutive bursts, illustrating the accuracy in resolving the time delay in the activation of the clusters. The two bursts contain the same clusters, but the activation sequences are slightly different. Blue dots mark the ignition time, and light green dots signal the clusters that fired simultaneously. The bottom orange boxes depict the final activation sequences of each burst. In the construction of the directed functional network, the influence of a cluster on another is conditioned by the time span between their activations. Close activations result in strong couplings (green arrows); far activations in weak ones (blue). Any two clusters whose activations are above 200 ms are considered functionally uncoupled (red).

and 9 of them form a subset that regularly fires together. The series of frames show the progress in cluster activation, revealed by the changes in fluorescence. Activity starts at the top-left cluster and progresses downwards. The time-line of sequence activation after image analysis is shown in Fig 4.3 (b), and the actual fluorescence traces are shown in Fig. 4.4. With our 10 ms temporal resolution we could resolve well the propagation of activity from a cluster to its neighboring ones (black arrows in Fig. 4.3 (b)). However, and for about 5% of the cases, the time delay between clusters' activation was either too short for detection or activation occurred simultaneously. The clusters associated to these 'simultaneous' events are marked in yellow in Fig. 4.3 (b), and their inter–relation was treated as a bi-directional link (yellow arrow), since no causality can be inferred.

A typical recording provided on the order of a 100 bursting episodes. Some of them included the same group of clusters, although the precise sequence of activation could vary. An illustrative example is shown in Fig. 4.4, which depicts the fluorescence traces of the 9 clusters along two consecutive bursts. The first sequence corresponds to the sketch of Fig. 4.3 (b). The orange box at the bottom of the plot indicates the relative activation time of each cluster within the window, with two clusters treated as simultaneous.

Measurements in 15 different clustered networks were carried out, in the following we will refer to them to the networks labeled 'A' – 'O'. In order to compare their properties with the ones from cultures with distinct structure, we applied the same measuring protocols and data analysis to 6 cultures characterized with a homogeneous distribution of neurons (obtained by limiting the mobility of neurons during growth), and labeled them as networks 'P' – 'U'.

### 4.1.2 CONSTRUCTION OF THE DIRECTED FUNCTIONAL NETWORKS

The above sequences of clusters' activation, extended to all the clusters and bursting episodes of the monitored culture, convey information on the degree of causal influence between any pair of clusters in the network. For instance, cluster #5 in Fig. 4.3 (b) can fire because of the first order influence of clusters #3 and #4 (see Fig. 4.4), but also because of the second and third order influences of clusters #2 and #1, respectively. Hence, a realistic functional network construction should take into account these possible influences from the upstream connected clusters to build a network whose links are not only directed, but also weighted by the

time delays in activation. This weighted treatment of the interaction between clusters is the major novelty of our work and the backbone of our model.

More formally, the interaction between any two clusters follows the principle of causality, i.e. the firing of cluster $j$ immediately after cluster $i$ eventually implies that cluster $i$ has induced the activity of $j$ at that particular time. The likelihood of this relation between clusters is weighted according to its frequency along the full observational time, allowing to an statistical validation. Indeed, cluster $i$ could induce the activity of various clusters, if all of them activate in a physically plausible short time window after cluster $i$. Such a construction is illustrated in Fig. 4.5.

To construct the directed functional networks for each studied culture we proceed as follows. First of all, we divide the entire firing sequence into the bursts of clusters' activity (Fig. 4.5 (A)) using the cut–off of 200 ms introduced in the previous section. Once the bursts have been detected, we compute the frequency distribution of time lags between pairs of consecutive firings. This frequency distribution informs about the characteristic times expected between two consecutive firings within the same burst, and hence it is a good proxy of the causal influence of a cluster on another. We will use this information to weight the causal influence of firing propagation. The frequency distribution presents a good fit to a universal Gaussian decay ($y \sim e^{-x^2/c}$) in all the analyzed cultures, although the variance $c$ is specific for each culture. The last step in the construction of the directed functional networks consists in linking the interactions within each burst, and weighting them according to the previous frequency distribution (Fig. 4.5 (B)). The rationale behind this process is as follows: we hypothesize that every cluster influences other clusters (posterior in time) within a burst and, the larger the time after a cluster has fired the lower the influence we expect in the activation of another cluster (simply because the signal fades out). Then, the weighting of the interaction by the expected frequency observed in the distribution conveys the functional influence between clusters. The weights are reinforced every time the same pair of clusters' sequence is observed. After processing the full sequence we obtain a peer-to-peer activation map that is our proxy of the functional network. We proceeded identically to construct the directed functional networks for homogeneous cultures, with the only difference that the cut-off time corresponds to 10 ms.

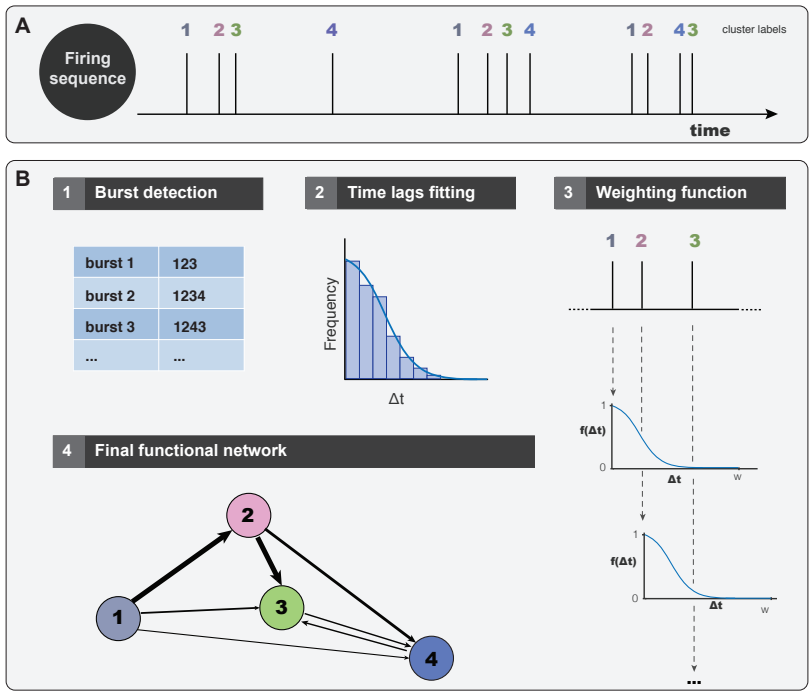



Figure 4.5: Sketch of the construction of the directed functional network. **A.** Schematic representation of the experimental data, with 12 firings of four different clusters (or neurons in the case of homogeneous cultures). **B.** Stages of the method to construct the directed functional network. (1) The first step consists in detecting the different bursts in the whole sequence. Firings that are separated by more than 200 ms for clustered cultures (10 ms for homogeneous ones) are not considered part of the same burst. (2) Calculation of the time lags between consecutive firings inside the bursts, for the whole sequence; Fitting of the frequency distribution to a Gaussian distribution; And calculation of the variance that will be specific for each culture. (3) Weighting procedure example for the first burst. Cluster #1 can activate #2 and #3. The weight of the links 1→2 and 1→3 depends on the time differences between clusters' activation, and is given by the function f($\Delta t$) found by the previous fitting. Hence, weight $w_{1\to 2} \simeq 0.6$, and $w_{1\to 3} \simeq 0$. Cluster #3 can be activated as well by cluster #2, with $w_{2\to 3} \simeq 0.2$. (4) Schematic representation of the resulting directed functional connectivity network. The width of the connections is proportional to the weight of the links.

#### 4.1.2.1 *Stability of the networks to the variation of the cut-off and variance of the weighting function*

We used two main parameters to quantitatively construct the directed functional network, namely the cut–off time for causality, and the variance $c$ of the Gaussian-like weighting function. In the following, we test the stability of the generated functions to the variation of these two parameters.

To generate the networks shown in this work, the cut–off time is set to 200 ms, two times the maximum measured time delay between consecutive activations. The importance of the cut–off is twofold: first, it serves the purpose of discriminating two successive bursting episodes; secondly, it is used to exclude individual firing events from an actual cascade of activations. Although these individual firings account for less than 2% of the total activations, they may occur in regions of the culture that are physically distant —though temporary close— from an actual sequence, and therefore they would add spurious, long–range functional connections to the network.

To examine whether the choice of the cut–off does or does not substantially affect the features of the generated functional network, we performed a sensitivity analysis on this parameter. As the process of generating the network from the sets of bursts is deterministic, we analyzed the influence of the cut–off value on the formed groups of firings. To quantify the variation on the bursts generated for different values of the cut–off, we calculated the variation of information [141] between the grouping of bursts obtained for different values of the cut–off to measure their difference (see Fig. 4.6). To generate this figure, the Variation of Information ($VI$) is computed as $VI(X,Y) = H(X) + H(Y) - 2I(X,Y)$, where $X$ and $Y$ are two partitions, $H$ is the entropy and $I$ is the mutual information. In our case, each partition is taken to be the set of bursts found at a certain value of the cut–off. We have screened the cut–off values from 0 to 1000ms. In the case of clustered cultures, we found that for values of cut–off of $200 \pm 50$ ms the variation of information is, on average, on the order of $10^{-2}$. In the homogeneous case, for cut–off values of $10 \pm 5$ ms, this value is on the order of $10^{-3}$. This means that varying the cut–off values in these regions does not substantially change the grouping of the bursts, and therefore the generated networks are equivalent.

On the other hand, the variance $c$ is obtained from a Gaussian fit of the distribution of consecutive activation delays within bursts. The value of $c$ is specific for each culture to take into account particular differences in the dynamics of the

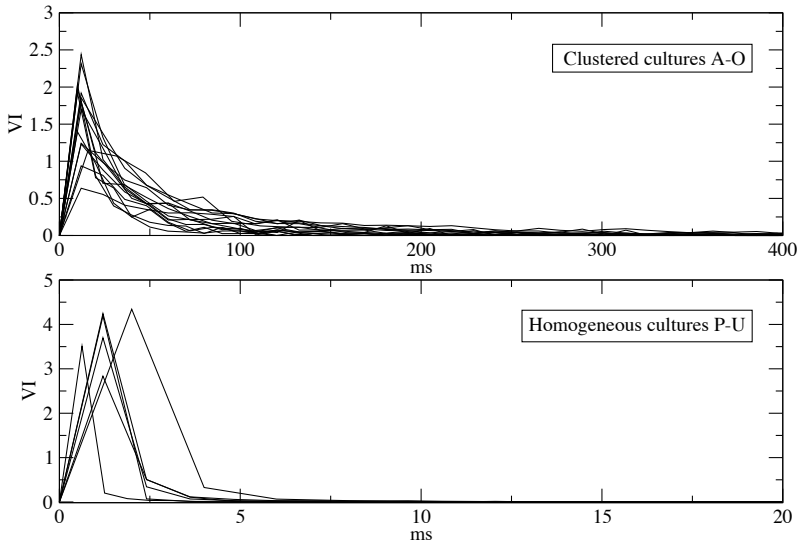



Figure 4.6: Sensitivity of the functional network construction to the cut-off times. The cut-off determines the end of a sequence and therefore its variation modifies the set of bursts. The cut-off is set to 200ms for clustered cultures and 10ms for homogeneous ones. To assess the sensitivity of the grouping of bursts to the cut-off, we have computed the Variation of Information (VI) between the grouping of bursts at a certain cut-off value and the previous one. Each line of the plot represents a different culture. As we can see, values of 10ms for homogeneous and 200ms for clustered cultures are values for which the variation of information is already stabilized.

network, specifically the number of *in vitro* days or the number of clusters (see Fig. 4.7), parameters that could affect the delay times of activation. Young cultures for instance exhibit longer time delays between pairs of clusters, leading to a distribution f($\Delta t$) shifted towards higher values and therefore a larger $c$. This reinforces the idea that a different variance $c$ has to be chosen for each culture.

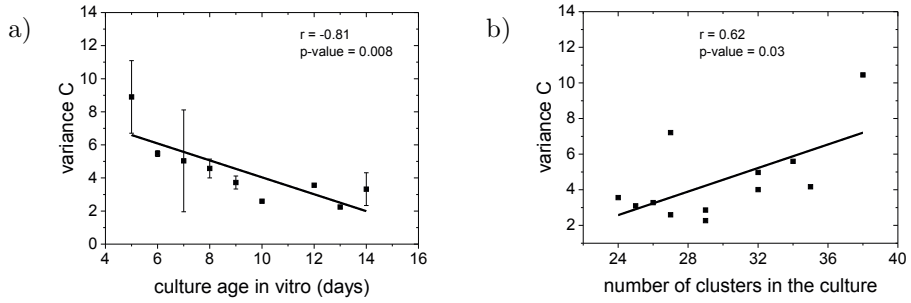



Figure 4.7: Dependence of the variance $c$ on the culture properties. **a)** The variance $c$ is obtained from the Gaussian fit of the activation delays between pairs of clusters. The plot shows that $c$ decreases with the culture age in vitro, indicating that young cultures display a slower dynamics (larger delay times and therefore larger variance) than mature cultures. Error bars show standard deviation. **b)** The variance increases with the number of clusters in the culture, indicating a broader and richer distribution of time delays as more clusters participate in the dynamics of the network.

#### 4.1.2.2 *Comparison with an alternative Mutual Information-based model for constructing the functional networks*

Additionally, to assess the goodness of our construction to infer the functional connectivity of the clustered networks, we compared our connectivity maps with those procured by information theoretic measures, such as Mutual Information or Transfer Entropy, applied to the original fluorescence recordings. These approaches have been used to draw the topological properties of neuronal networks *in vitro*, both in electrode recordings [21, 84] and calcium fluorescence imaging [203, 166]. The comparison of our method with these theoretic measures showed that the identified functional links were fundamentally the same, with small quantitative differences associated to the particular weighting procedures. In Appendix A.2 we present a detailed explanation of the alternative method proposed to construct the networks, based on Mutual Information.

### 4.1.3 ANALYSIS OF THE FUNCTIONAL NETWORKS

We computed the functional networks of the 15 realizations of clustered cultures ('A' to 'O'), as well as the 6 homogeneous ones ('P' to 'U'), and analyzed some major topological traits. Firstly, for each culture we obtained the number of nodes, the number of edges, the average degree of the networks, and its average strength. These topological measures are summarized in Table 4.1. Although young cultures display a richer activity, in general all networks present a similar number of nodes and a comparable functional connectivity, which is described by the number of edges, the average degree and the average strength.

Representative examples of the investigated functional networks for the clustered configuration are shown in Fig. 4.8. The position of the nodes and their size are the same as the actual clusters for easier comparison. Edges in the directed network are both color and thickness coded to highlight their importance, with darker colors corresponding to the highest weights. This representation reveals those pairs of clusters that maintain a persistent causality relationship over time. Nodes are also color coded according to their strength, i.e. the total weight of the inwards and outwards edges.

The functional networks exhibit some interesting features. First, there are groups of nodes that form tightly connected communities. These communities actually reflect the most frequent bursting sequences. Second, nodes preferentially connect to neighboring ones with some long–range connectivity, and often following paths that are not the major physical connections. This indicates that the structural connectivity of the network cannot be assessed from just an examination of the most perceivable processes. And third, as shown in Fig. 4.9, we observed that there is no correlation ($R = 0.040$, $p = 0.88$) between the width of the physical connections and their corresponding weight in the functional networks, or the size of the nodes and their strength ($R = 0.072$, $p = 0.70$, (see Fig. 4.10), and indicates that the dynamical traits of the network cannot be inferred from its physical configuration, stressing the importance of the functional study.

We also observed that the size of the clusters did not correlate with their average activity ($R = 0.14$, $p = 0.51$, see Fig. 4.11), i.e. small and big clusters displayed similar firing frequencies, and of 1 firing/min on average. However, since some clusters are initiators of activity and others just followers, we also computed the relative contribution of a given cluster size to initiate activity in the

| Culture type | Network | DIV | burst rate ($min^{-1}$) | Number of nodes | Number of edges | Average degree | Average strength |
|---|---|---|---|---|---|---|---|
| Clust. | **A** | 5 | 7.35 | 38 | 544 | 14.32 | 33.71 |
| | **B** | 6 | 6.80 | 34 | 1044 | 30.71 | 131.94 |
| | **C** | 6 | 6.55 | 29 | 762 | 26.28 | 142.45 |
| | **D** | 7 | 2.99 | 27 | 471 | 17.44 | 32.53 |
| | **E** | 7 | 0.87 | 29 | 660 | 22.76 | 22.83 |
| | **F** | 8 | 0.85 | 32 | 750 | 23.44 | 12.89 |
| | **G** | 8 | 0.79 | 35 | 395 | 11.29 | 4.99 |
| | **H** | 9 | 1.17 | 32 | 722 | 22.56 | 30.99 |
| | **I** | 10 | 3.19 | 27 | 486 | 18.00 | 81.88 |
| | **J** | 12 | 2.42 | 24 | 456 | 19.00 | 97.32 |
| | **K** | 13 | 1.28 | 19 | 252 | 13.26 | 38.27 |
| | **L** | 14 | 3.40 | 17 | 116 | 6.82 | 28.25 |
| | **M** | 14 | 1.40 | 25 | 205 | 8.20 | 16.71 |
| | **N** | 14 | 1.86 | 26 | 437 | 16.81 | 30.63 |
| | **O** | 14 | 4.91 | 29 | 391 | 13.48 | 38.77 |
| Homo. | **P** | 6 | 3.30 | 814 | 453812 | 557.51 | 41.24 |
| | **Q** | 8 | 4.10 | 589 | 243606 | 413.59 | 27.66 |
| | **R** | 10 | 0.27 | 562 | 35379 | 62.95 | 2.46 |
| | **S** | 15 | 1.05 | 1107 | 239517 | 216.37 | 9.47 |
| | **T** | 16 | 0.47 | 694 | 274278 | 395.21 | 24.89 |
| | **U** | 16 | 0.78 | 703 | 155643 | 221.40 | 10.34 |

Table 4.1: Network features of clustered and homogeneous cultures.

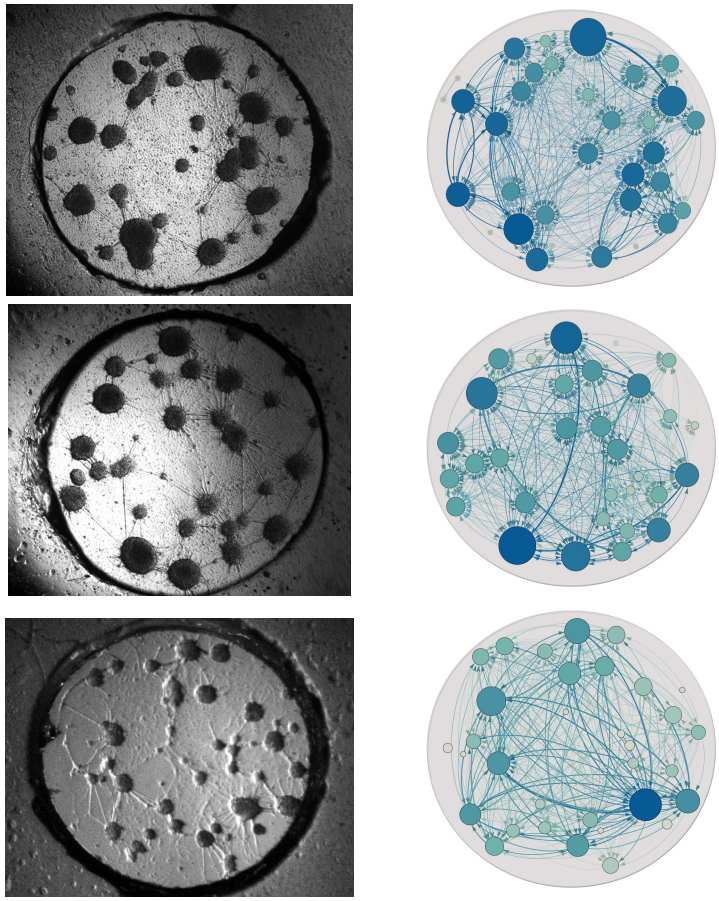


Figure 4.8: Clustered neuronal cultures and their corresponding functional networks. Left column: Bright field images of 3 representative neuronal cultures at different *days in vitro*. Right column: Corresponding functional networks obtained from the directed and weighed construction described in Fig. 4.5. Downwards, the pictures correspond to the cultures labeled D, H and O in Table 4.1. Only active clusters are used in the construction of the functional network. The size of the nodes is similar to the ones observed in the cultures, and facilitates the visual comparison of the functional network with the real culture. In the functional networks, the nodes and edges are both color and thickness coded according to their strength and weight, respectively. The darker the color, the higher the value.

network. We found no significant correlation between initiation and cluster size ($R = 0.38$, $p = 0.14$, see Fig. 4.12). These results strengthen the conclusion that one cannot predict the clusters that will initiate activity, or the most persistent sequences, by just a visual inspection of cluster sizes and their distribution over the network.

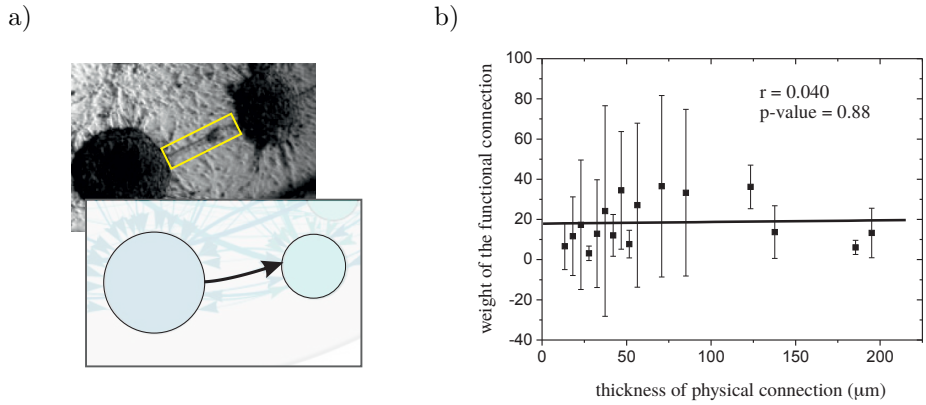



Figure 4.9: Dependence of the weight of the functional connections on the width of physical connections between directly connected clusters. **a)** Conceptual sketch to show the comparison between structural and functional links. **b)** Analysis of 102 pairs of clusters, with data binned for similar widths. No significant correlation is observed between the actual thickness of the physical links and the weight of the functional connections.

### 4.1.4 ASSORTATIVITY COEFFICIENTS OF THE FUNCTIONAL NETWORKS

To further characterize the structure of the functional networks created, we are interested in calculating the assortativity coefficients and the rich–club properties. The *assortativity* measure was introduced by Newman in [149] and is a global quantity that measures the preference of high-degree vertices to attach to other high-degree vertices. If this happens, then we say that the network shows *assortative mixing*. On the contrary, networks can also show *disassortative mixing*, meaning that high-degree nodes tend to connect with low-degree nodes. Networks can also present no assortative or disassortative mixing, which are then considered as *neutral*. Assortative networks have been observed in both structural [101] and functional [67] human brain networks. It has been proposed that assortative

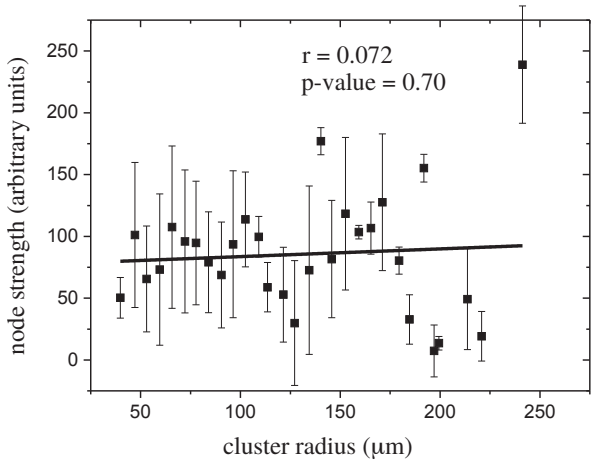



Figure 4.10: Plot of the strength of each functional node as a function of its size. The dependence of the node strength on cluster size shows no correlation, indicating that the functional connectivity cannot be assessed from the size of the clusters. Data is based on the analysis of 537 clusters.

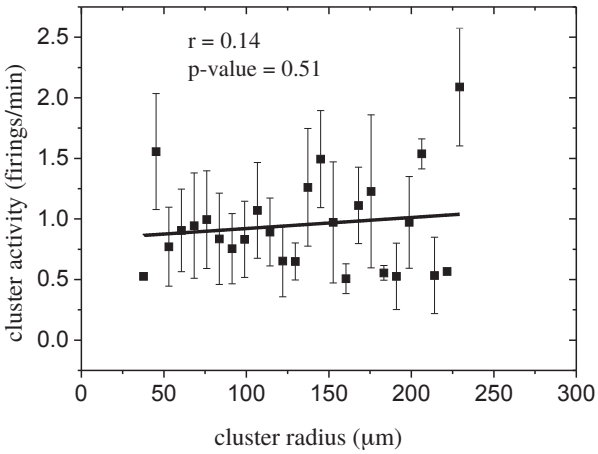



Figure 4.11: Plot of the volume of activity of a cluster as a function of its size, for the same clusters of Fig. 4.10. No correlation is observed.

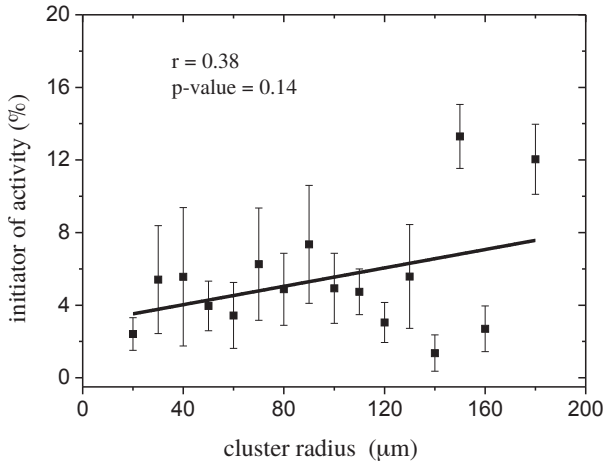



Figure 4.12: Activity within a burst is always initiated by a particular cluster, which triggers the sequential activation of all the downstream clusters. To quantify the importance of these *initiators of activity* in the network dynamics we computed the number of times that a cluster of a given size initiates a sequence of activations. The plot shows that there is not a significant correlation between initiation and size. The analysis is based on the study of 1800 bursts. All these results indicate that the functional connectivity cannot be drawn from a visual inspection of the neuronal culture. Error bars show standard deviation.

networks exhibit a modular organization [11], display an efficient dynamics that is stable to noise [56], and manifest resilience to node deletion (either random or targeted) [184, 149]. Resilience is ascribed to the preferred interconnectivity of high–degree nodes, which shape a 'connectivity backbone' [1] that preserves network integrity. On the other hand, disassortative networks, such as the ones identified in the yeast's protein–protein interaction and the neuronal network of *C. elegans* [149], are more vulnerable to targeted attacks. However, in these disassortative networks, the tendency of high degree nodes to connect with low degree ones results in a star–like topology that favors information processing across the network.

The assortativty coefficient is usually calculated through the Pearson correlation coefficient between the unweighted degrees of each link in the network [149]. To account for effects associated to large networks, the Spearman assortativity measure was introduced [135] and, later, weighted assortativity measures were proposed to include the weight in degree–degree dependencies [134].

To be able to calculate the assortativity coefficient of the weighted functional neuronal networks generated, we propose a new measure of assortativity that explicitly incorporates the weight of the links. We observed that all the studied functional networks derived from clustered cultures show a strong, positive assortative mixing that is maintained along different stages of development. On the contrary, homogeneous cultures tend to be weakly assortative, or neutral. Finally, in combination with experiments that measure the robustness of network activity to circuitry deterioration, we show that the strongly assortative, clustered networks are more resistant to damage compared to the weakly assortative, homogeneous ones. This provides a prominent example of the existence of assortativity in biological networks, and illustrates the utility of clustered neuronal cultures to investigate topological traits and the emergence of complex phenomena, such as self-organization and resilience, in living neuronal networks.

#### 4.1.4.1 *Generalization of assortativity to directed weighted networks*

Newman [149] defined *assortativity* $\rho^P$ as the Pearson correlation between the degrees of every pair $E$ of linked nodes in the network. More precisely, in the case of directed networks, if $k_i^{\text{out}} = \sum_j a_{ij}$ is the output degree from node $i$,

$k_j^{\text{in}} = \sum_i a_{ij}$ the input degree to node $j$, and $E$ scans all the edges in the network, then

$$\rho^P = \frac{\frac{1}{L}\sum_{(i,j)\in E}\left(k_i^{\text{out}} - \langle k^{\text{out}}\rangle_E\right)\left(k_j^{\text{in}} - \langle k^{\text{in}}\rangle_E\right)}{\sqrt{\frac{1}{L}\sum_{(i,j)\in E}\left(k_i^{\text{out}} - \langle k^{\text{out}}\rangle_E\right)^2}\sqrt{\frac{1}{L}\sum_{(i,j)\in E}\left(k_j^{\text{in}} - \langle k^{\text{in}}\rangle_E\right)^2}}, \tag{4.1}$$

where

$$\langle k^{\text{out}}\rangle_E = \frac{1}{L}\sum_{(i,j)\in E} k_i^{\text{out}} = \frac{1}{L}\sum_{i=1}^{N}\sum_{j=1}^{N} a_{ij}k_i^{\text{out}} = \frac{1}{L}\sum_{i=1}^{N}(k_i^{\text{out}})^2, \tag{4.2}$$

$$\langle k^{\text{in}}\rangle_E = \frac{1}{L}\sum_{(i,j)\in E} k_j^{\text{in}} = \frac{1}{L}\sum_{i=1}^{N}\sum_{j=1}^{N} a_{ij}k_j^{\text{in}} = \frac{1}{L}\sum_{j=1}^{N}(k_j^{\text{in}})^2, \tag{4.3}$$

$$L = \sum_{(i,j)\in E} a_{ij} = \sum_{i=1}^{N}\sum_{j=1}^{N} a_{ij}. \tag{4.4}$$

After some algebra, assortativity may also be written as

$$\rho^P = \frac{\sum_{i,j} a_{ij}\left(k_i^{\text{out}} - \langle k^{\text{out}}\rangle_E\right)\left(k_j^{\text{in}} - \langle k^{\text{in}}\rangle_E\right)}{\sqrt{\sum_{i,j} a_{ij}\left(k_i^{\text{out}} - \langle k^{\text{out}}\rangle_E\right)^2}\sqrt{\sum_{i,j} a_{ij}\left(k_j^{\text{in}} - \langle k^{\text{in}}\rangle_E\right)^2}} \tag{4.5}$$

$$= \frac{\sum_{i,j} a_{ij}\left(k_i^{\text{out}} - \langle k^{\text{out}}\rangle_E\right)\left(k_j^{\text{in}} - \langle k^{\text{in}}\rangle_E\right)}{\sqrt{\sum_{i} k_i^{\text{out}}\left(k_i^{\text{out}} - \langle k^{\text{out}}\rangle_E\right)^2}\sqrt{\sum_{j} k_j^{\text{in}}\left(k_j^{\text{in}} - \langle k^{\text{in}}\rangle_E\right)^2}} \tag{4.6}$$

This definition of assortativity is applicable to all kind of networks, either undirected, directed, weighted or unweighted. For weighted networks as in our case, the strength of the nodes carries important information about the structure of the network, and thus it would be useful to know the correlation between the strengths instead of the degrees. Since in this case each edge carries a weight, it seems logical that edges with higher weight should have a larger contribution to the correlation. Therefore, we define the *weighted assortativity* $\rho^{PW}$ as the Pearson weighted correlation between the strengths of the nodes. In mathematical

terms, if $w_{ij}$ is the weight of the link from node $i$ to node $j$ (zero if there is no link), then $s_i^{\text{out}} = \sum_j w_{ij}$ and $s_j^{\text{in}} = \sum_i w_{ij}$ are the output and input strengths, then

$$\rho^{PW} = \frac{\sum_{i,j} w_{ij} \left(s_i^{\text{out}} - \langle s^{\text{out}} \rangle_E\right) \left(s_j^{\text{in}} - \langle s^{\text{in}} \rangle_E\right)}{\sqrt{\sum_i s_i^{\text{out}} \left(s_i^{\text{out}} - \langle s^{\text{out}} \rangle_E\right)^2} \sqrt{\sum_j s_j^{\text{in}} \left(s_j^{\text{in}} - \langle s^{\text{in}} \rangle_E\right)^2}}, \tag{4.7}$$

where

$$\langle s^{\text{out}} \rangle_E = \frac{1}{S} \sum_{(i,j)\in E} w_{ij} s_i^{\text{out}} = \frac{1}{S} \sum_i (s_i^{\text{out}})^2, \tag{4.8}$$

$$\langle s^{\text{in}} \rangle_E = \frac{1}{S} \sum_{(i,j)\in E} w_{ij} s_j^{\text{in}} = \frac{1}{S} \sum_j (s_j^{\text{in}})^2, \tag{4.9}$$

with $S$ the total strength of the network, i.e.

$$S = \sum_{(i,j)\in E} w_{ij} = \sum_{i,j} w_{ij}. \tag{4.10}$$

Litvak *et al.* [135] showed that in disassortative networks the magnitude of the standard assortativity decreases with network size, a problem that was solved by replacing the Pearson correlation $\rho^P$ with the Spearman correlation, thus obtaining a *Spearman assortativity* $\rho^S$. Spearman rank correlation is calculated in the same way that the Pearson correlation but substituting the values (in this case, the degrees of the nodes) by their respective ranks, i.e. their position when the values are sorted in ascending order. This leads us to define the *Spearman weighted assortativity* $\rho^{SW}$ as the Spearman weighted correlation between the strengths of the nodes at both ends of each edge in the network.

The estimation of the error in the assortativity value (for any of the previous four variants) can be computed in several ways, for instance through the jackknife method, the bootstrap algorithm, or by using the Fisher transformation [150, 66]. We used in our work the bootstrap algorithm, and considered 1000 random samples of the data.

#### 4.1.4.2 *Assortativity values of the functional neuronal networks*

We determined the values of the weighted formulation of assortativity, both for the Pearson $\rho^{PW}$ and Spearman $\rho^{SW}$ correlations, with values in range $[-1, 1]$.

| Culture type | Network | Assortativity Pearson (wh) | Assortativity Spearman (wh) |
|---|---|---|---|
| Clust. | **A** | 0.642 ± 0.044 | 0.605 ± 0.036 |
| | **B** | 0.404 ± 0.040 | 0.449 ± 0.035 |
| | **C** | 0.440 ± 0.036 | 0.425 ± 0.041 |
| | **D** | 0.442 ± 0.064 | 0.414 ± 0.063 |
| | **E** | 0.402 ± 0.043 | 0.396 ± 0.045 |
| | **F** | 0.317 ± 0.064 | 0.287 ± 0.057 |
| | **G** | 0.528 ± 0.086 | 0.553 ± 0.063 |
| | **H** | 0.355 ± 0.061 | 0.377 ± 0.052 |
| | **I** | 0.460 ± 0.046 | 0.478 ± 0.046 |
| | **J** | 0.326 ± 0.049 | 0.322 ± 0.046 |
| | **K** | 0.729 ± 0.062 | 0.699 ± 0.051 |
| | **L** | 0.586 ± 0.076 | 0.552 ± 0.077 |
| | **M** | 0.356 ± 0.084 | 0.309 ± 0.087 |
| | **N** | 0.698 ± 0.060 | 0.664 ± 0.061 |
| | **O** | 0.372 ± 0.069 | 0.364 ± 0.061 |
| Homo. | **P** | 0.059 ± 0.005 | 0.055 ± 0.005 |
| | **Q** | 0.038 ± 0.007 | 0.036 ± 0.007 |
| | **R** | 0.112 ± 0.023 | 0.125 ± 0.023 |
| | **S** | 0.111 ± 0.008 | 0.107 ± 0.008 |
| | **T** | 0.077 ± 0.007 | 0.067 ± 0.006 |
| | **U** | 0.040 ± 0.009 | 0.037 ± 0.009 |

Table 4.2: Assortativity values of the clustered and homogeneous cultures, for the weighted Pearson and Spearman formulations of assortativity. As we can see, clustered networks display assortative mixing, while homogeneous ones display a very low or even neutral assortative mixing.

In Table 4.2 we can observe that all clustered networks (labeled 'A'-'O') exhibit a positive weighted assortativity, in the range $0.32 \leq \rho^{PW} \leq 0.73$ for the Pearson construction and $0.29 \leq \rho^{SW} \leq 0.70$ for the Spearman one. Although the values fluctuate across different cultures, the two assortativity measures provide the same value within statistical error, and reflect that network size corrections provided by the Spearman's treatment have little influence in strongly assortative networks.

Interestingly, for the experiments with a homogeneous distribution of neurons (labeled 'P' − 'U'), the assortativity values are much lower (by an order of magnitude on average) than the ones for clustered cultures, in the range $0.04 \leq \rho^{PW} \leq 0.11$ for Pearson's and $0.04 \leq \rho^{SW} \leq 0.12$ for Spearman's.

### 4.1.5 RICH–CLUB PROPERTIES

To assess the importance of the measured assortativity values, we have also computed the degree of *rich–clubness* of the generated networks. The rich–club phenomenon refers to the tendency of nodes with high degree to form tightly interconnected communities, compared to the connections that these nodes would have in a null model that preserves the node's degree but otherwise is totally random [223]. The calculation of the rich–club allows us to reinforce the assortativity analysis presented before. Assortative mixing, in principle, will induce a rich–club effect that should be clearly detectable for a wide range of degrees (or weights).

#### 4.1.5.1 *Calculation of the rich–club for weighted directed networks*

The rich–club analysis computes the degree–degree (or weight–weight) correlation distributions, respect to a null case of non–correlated degrees (or weights). The weighted formulation for the rich–club takes into account the node's strength instead of the degree, and is particularly useful in situations in which the weights of the links can not be overlooked. We will first introduce the formulation for the calculation of rich–club in weighted networks as presented in [192], and afterwards we will extend it to the case of weighted directed networks.

The rich–club score $\phi_{s_T}^{\text{unc}}$ relative to the uncorrelated null case is calculated as follows:

$$\phi_{s_T}^{\text{unc}} = \frac{W_{s_T}}{W_{s_T}^{\text{unc}}}, \tag{4.11}$$

where $W_{s_T}$ is the sum of the weights of the links of the subgraph formed only by those nodes whose strengths are higher than $s_T$,

$$W_{s_T} = \sum_{i \in v_{s_T}} \sum_{j \in v_{s_T}} w_{ij}, \tag{4.12}$$

and $W_{s_T}^{\text{unc}}$ is the corresponding value in the case of uncorrelated strengths:

$$W_{s_T}^{\text{unc}} = \langle s \rangle \frac{\sum_{i \in v_{s_T}} \sum_{j \neq i \in v_{s_T}} s_i s_j}{N \langle s \rangle^2 - \langle s^2 \rangle}. \tag{4.13}$$

The term $v_{s_T}$ designates the subset of nodes $i$ such that $s_i > s_T$. $N$ is the total number of nodes in the network, and $\langle s \rangle$ and $\langle s^2 \rangle$ are the first and second moments of the strength distribution. The ratio $\phi_{s_T}$ is calculated for all values of $s_T$, and ranges from the minimum value of strength in the network to the maximum. This ratio indicates the presence or absence of a rich–club in the network: a network shows a rich–club effect when the high values of $s_T$ give a ratio above 1.

To calculate this ratio on our functional networks, we have to consider that the network is not only weighted but also directed. Therefore, we need to adapt the former formulation for the case of weighted directed networks. For this reason we will consider the in– and out–strength of each node, expressed as $s_i^{\text{in}}$ and $s_i^{\text{out}}$ respectively. In the directed formulation, Eqs. 4.11 and 4.12 remain unchanged. However, $v_{s_T}$ must be redefined as

$$v_{s_T} = \{i \,|\, s_i^{\text{in}} + s_i^{\text{out}} > 2s_T\}, \tag{4.14}$$

and the term $W_{s_T}^{\text{unc}}$ becomes

$$W_{s_T}^{\text{unc}} = \langle s \rangle \frac{\sum_{i \in v_{s_T}} \sum_{j \neq i \in v_{s_T}} s_i^{\text{out}} s_j^{\text{in}}}{N \langle s \rangle^2 - \langle s^{\text{out}} s^{\text{in}} \rangle}, \tag{4.15}$$

where the averages are calculated as

$$\langle s \rangle = \frac{1}{N} \sum_i s_i^{\text{out}} = \frac{1}{N} \sum_i s_i^{\text{in}}, \quad \langle s^{\text{out}} s^{\text{in}} \rangle = \frac{1}{N} \sum_i s_i^{\text{out}} s_i^{\text{in}}. \tag{4.16}$$

This formulation allows us to calculate the rich–club coefficient for weighted directed networks. Note that for an undirected network (where $s_i^{\text{in}} = s_i^{\text{out}}$) the latter formulation reduces to the original one.

#### 4.1.5.2 *Rich–club analysis of the functional neuronal networks*

The evaluation of the rich–club $\phi^{\text{unc}}(s_T)$ is performed by computing the ratio between the connectivity strength of highly connected nodes and its randomized counterpart, for gradually higher values of the strength threshold $s_T$. Ratios larger than 1 indicate that higher strength nodes are more interconnected to each

other than what one would expect in a random configuration. On the contrary, a ratio less than 1 reveals an opposite organizing principle that leads to a lack of interconnectivity among high–degree nodes. We expect the results of the rich–club to reinforce the positive or neutral assortativity found for the case of the clustered and homogeneous cultures, respectively. In principle, networks with high assortativity coefficient should present rich–club ratios larger than 1 for a long range of values of $s_T$, while neutral assortativity networks should show little or no rich–club organization.

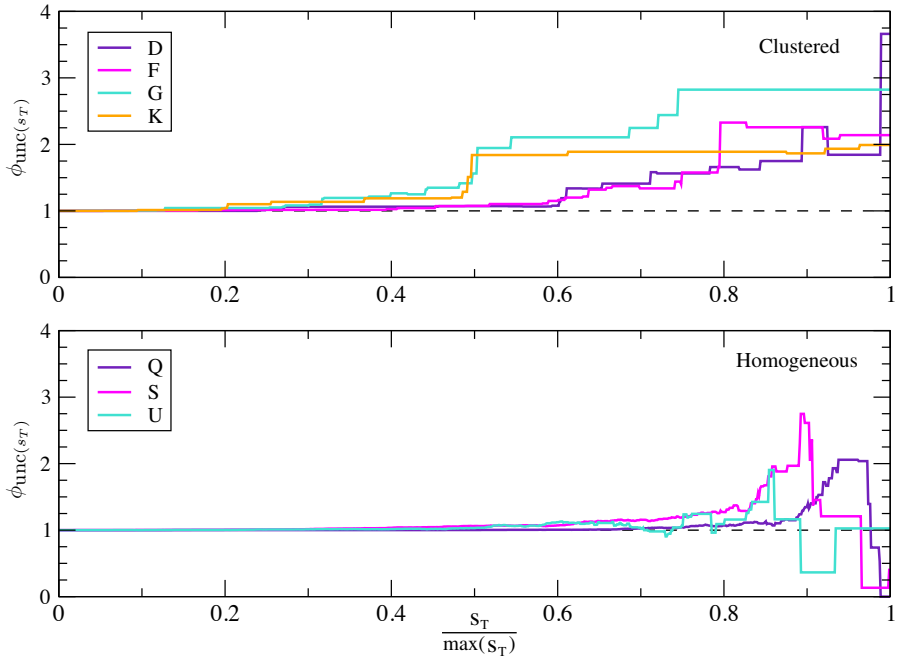



Figure 4.13: Rich–club ratio as a function of the strength threshold $s_T$ for a sample of four clustered cultures and three homogeneous ones. We observe clustered cultures having larger rich–club ratios than the homogeneous ones, and also sustained through larger ranges of $s_T$.

Indeed, after the calculation of the ratios for the studied clustered networks, we found a positive tendency towards the creation of rich–clubs in all of them, while homogeneous networks have rich–club ratios larger than 1 only for a short range of values of $s_T$. In Fig. 4.13 we plot the rich–club ratio as a function of

the strength threshold $s_T$ for four clustered networks and three homogeneous ones. As we can see, clustered networks start to present rich–clubs much before the homogeneous ones, and the positive ratios of rich–club are sustained until the maximum value of $s_T$ is reached. On the contrary, homogeneous networks present spurious peaks of rich–clubness, alternated with periods in which the rich–club is non existent or even negative.

### 4.1.6 EXPERIMENTAL CHECK OF THE NETWORK RESILIENCE

Several studies highlight the importance of assortative features for network resilience to damage. Given the strong assortativity of our clustered cultures, we carried out a new set of experiments to investigate the concurrent presence of resilient traits. To do so, we considered two major 'damaging' actions to the network. In a first one, we gradually weakened the excitatory network connectivity by means of the AMPA–glutamate antagonist CNQX, and measured the decay in spontaneous activity as connectivity failed. In a second one, we continuously exposed a culture to strong fluorescence light, therefore inducing photo-damage to the neurons. This action resulted in random neuronal death across the network and hence a progressive failure of its spontaneous dynamics. The rate of activity decay upon radiation damage provided an estimation of the resistance of the network to node deletion. These investigations were carried out at the same time in clustered cultures (strongly assortative) and in homogeneous ones (weakly assortative or neutral). Their comparison provides a first reference to relate assortativity, network topology and resistance to damage.

Fig. 4.14 (top) shows the results for the application of CNQX to a clustered culture. We first monitored each cluster individually in the unperturbed case, and measured its average firing activity $\gamma_0$ along 15 min. We then applied a given drug concentration, measured the firing activity $\gamma$ for another 15 min, and computed the relative changes in activity respect to the unperturbed case, as $\Gamma \equiv (\gamma - \gamma_0)/\gamma_0$. The protocol was repeated until activity ceased. Two illustrative examples of the action of CNQX on network activity are provided in Fig. 4.14. In a clustered cultured and for weak CNQX applications ($\simeq 100$ nM) the activity in some clusters increases, while in some other decreases, and on average the network firing rate remains stable ($\Gamma \simeq 0$). As CNQX is increased to 600 nM, we observe that most of the clusters have reduced their activity, although there are still some that maintain a high activity or even increase it. This different behavior from

cluster to cluster suggests that clustered networks are highly flexible, and that they may have mechanisms to preserve activity even with strong weakening of the connectivity. Conversely, homogeneous cultures (Fig. 4.14 (bottom)) lose activity in a more regular and faster way. Homogeneous networks are characterized by a highly coherent dynamics [165], and therefore all neurons in the network reduce activity similarly as CNQX is applied. Interestingly, for $[\mathrm{CNQX}] \simeq 400$ nM the shown homogeneous culture has almost completely silenced ($\Gamma \simeq -1$), a value of CNQX for which the clustered culture is still highly active. We repeated this study on 4 different realizations of each culture type and observed that, on average, the critical concentration $[\mathrm{CNQX}]_\mathrm{C}$ at which activity complete stopped was 1.6 $\mu$M for clustered and 0.5 $\mu$M for homogeneous networks (Fig. 4.15 (a)).

Fig. 4.15 (b) shows the results for the resistance of the networks to node deletion as a consequence of direct photo–damage to the neurons. As can be observed, homogeneous cultures decay in activity much faster than the clustered ones, pinpointing the general resistance of clustered cultures to structural failure.

Summarizing, the complex networks toolset has proven to be an adequate approach to understand the functionality of cultured networks of neurons. The functional networks associated to the clustered cultures show positive assortative values and a positive tendency towards a rich–club structure. We hypothesize that the preference of clusters to connect with clusters with similar functional connectivity follows an intrinsic survival mechanism.

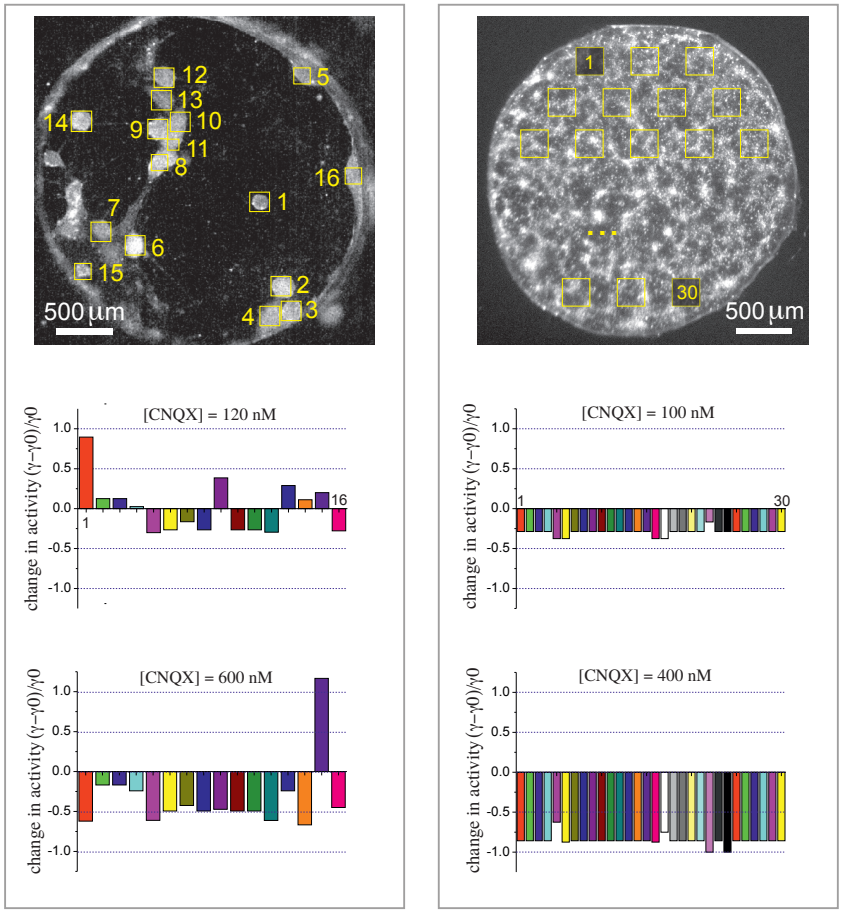



Figure 4.14: Examples of the degradation of neuronal activity in clustered and homogeneous cultures due to the gradual weakening of excitatory connectivity. Both culture types were investigated at the same *day in vitro* 14 and contained a similar density of neurons. The weakening of connections is achieved by gradually increasing the concentration of CNQX, an AMPA-glutamate receptor antagonist in excitatory neurons. Network response upon weakening is quantified through the relative change in activity $(\gamma - \gamma_0)/\gamma_0$ between a given CNQX application and the unperturbed state. Activity variations are indicated separately for each cluster, and shown according to the cluster labeling number. **Left column.** Clustered cultures show a mixed response upon weakening, with some clusters increasing activity and others reducing it. It is only for relatively high concentrations of CNQX ($\geq$ 600 nM) that the activity systematically decays up to the full silencing of the network. **Right column.** In homogeneous cultures, activity is analyzed in 30 regions that cover in a regular manner the entire network. Activity decays almost equally in all regions. Relatively small drug concentrations of CNQX $\simeq$ 400 nM practically suffice to fully stop activity.

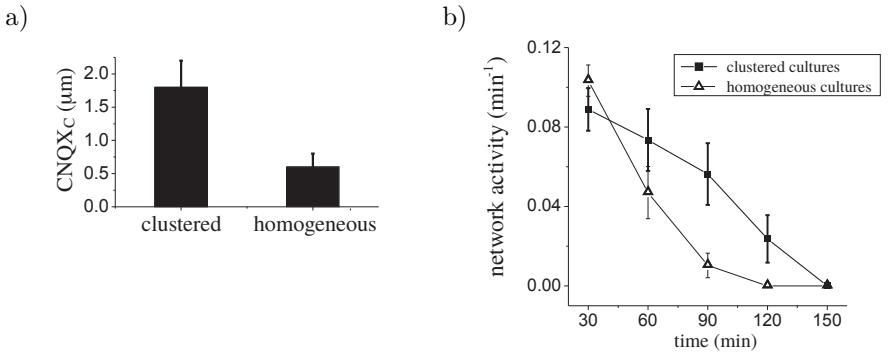



Figure 4.15: **a)** Average critical concentration $[\mathrm{CNQX}]_C$ at which spontaneous activity completely ceases, about 1.6 $\mu$M for clustered networks and 0.5 $\mu$M for homogeneous ones. Data is averaged over 4 network realizations of each type of culture. **b)** Photo–damage experiments. Spontaneous activity is measured in cultures that are continuously exposed to strong fluorescence light, causing gradual neuronal degradation and ultimately the death of the entire network. The total radiation received by the neurons is calculated as the duration of the exposure times the area covered by the neurons in the culture (1.9 $\mathrm{mm}^2$ and 2.3 $\mathrm{mm}^2$ on average for clustered and homogeneous clusters, respectively). The spontaneous activity in homogeneous cultures decays at a much faster rate than in the clustered counterparts. Data is averaged over 6 network realization of each type. Error bars show standard deviation.

UNIVERSITAT ROVIRA I VIRGILI
FROM COMMUNITY STRUCTURE TO THE PHYSICS OF MULTIPLEX NETWORKS
Clara Granell Martorell

# 5

# CONCLUSIONS AND FUTURE PERSPECTIVES

## CONCLUSIONS

During the course of this document, I have presented examples of complex systems of different nature, and shown the convenience of approaching their analysis through the complex networks toolset. We have analyzed synthetic and real networks, and even those extracted from an experimental biological system. We have studied the structure of networks as well as the dynamics in multiplex networks, and the main conclusions that can be extracted are summarized next.

Analyzing the modular structure of networks is very challenging. The definition of what is a community often depends on the particular setup that we analyze, and even when we are able to translate the chosen definition into a formal expression, the impossibility of exploring the whole space of partitions makes this problem also a computationally challenging one. Despite the difficulties, it is still a problem worth to be explored, mainly because the modular structure that we can discover is not only informative of merely topologically cohesive structures, but is also in many cases related to the functionality of the network.

The first problem that we encounter when we wish to do some sort of community detection is choosing a suitable algorithm for our purpose. There are plenty of methods in the literature, and we should be aware that choosing a particular algorithm means accepting the implicit definition of community that comes with it. We should also consider that each method is designed to operate on top of a certain type of data, and we should be careful of choosing an algorithm that respects the nature of the data that we wish to analyze. If for instance, our network is weighted, signed or directed, or even bipartite or time-

varying, we must go for an algorithm whose formulation is able to deal with such features. In this document I presented an application of a community detection algorithm which implied detecting groups of nodes on a network that had positive and negative weights. The method of choice, the AFG algorithm, didn't allow for negative weights, and therefore it had to be extended to the case of signed networks, following the intuition that negative edges should contribute to place nodes in different communities. With the appropriate formulation at hand, we were able to analyze the signed network without having to drop out the valuable information that signs provided us, thus obtaining the division in communities that we were looking for. This new formulation of the algorithm enables us to approach community detection analysis in many other scenarios, e.g. in correlation networks.

In fact, community detection algorithms are not only suited to obtain the modular structure of networks, they are versatile tools that can be used to address a variety of problems. Indeed, the extended version of the AFG method allowed us to approach a classical problem in computer science: the unsupervised classification of data, or data clustering. With the signed version of the AFG at hand, we decided to approach the problem of the classification of the Iris dataset using a complex networks approach. To do so, we computed a similarity network, where nodes account for the samples we wish to classify and links account for the *signed* similarity between them. Applying the AFG community detection algorithm to this network we were able to obtain a grouping of the data, with a success comparable to other well-known data clustering techniques.

Our algorithm of choice, the AFG algorithm, falls in the category of multi-resolution community detection algorithms, meaning that it is able to screen the whole mesoscale of the network and obtain a partition for each scale of resolution. Aside from the convenience that multi-resolution algorithms suppose in terms of being able to access the whole mesoscale —instead of obtaining results for a single, arbitrary scale—, this algorithm was designed this way in order to escape from the resolution limit of modularity. This limit provokes that certain configurations of communities, specially when the sizes of the groups are very small compared to the rest of the network, remain undetectable to modularity, something that —although expected— is inconvenient. Multi-resolution algorithms, by having access to the full mesoscale, are able to palliate this flaw of modularity. However, modularity suffers from another kind of problem, which we referred to in this document as the splitting and merging of communities. This problem is encountered when the communities in the network have very different sizes,

and is caused by the use of a single resolution parameter for the whole network. To solve this problem, we introduced the hierarchical version of the AFG algorithm, a method specially designed to correctly identify differently sized modules by using a resolution parameter for each module of the network, and proceeding hierarchically with each community found. As shown, this method is able to overcome the splitting and merging problem of modularity, even for the case where the sizes of the communities are dramatically different.

A common problem when approaching the challenge of detecting communities on a network is the assessment of the quality of the results obtained. After applying the algorithm of choice the user is provided with a partition —or set of partitions— without further information about the reliability of the grouping obtained, other than the belief that the algorithm does what it claims. To evaluate and compare different algorithms, it is common to use benchmarks, namely toy networks with an imposed community structure that we expect to recover when applying the algorithm. In the literature there are multiple benchmark generating tools, which have different purposes, such as generating networks with overlapping community structure or generating networks with communities embedded in different resolutions. A relatively recent problem is the challenge of detecting communities in time-varying networks, a topic provided with multiple algorithms to such purpose, but lacking tools in the validation part. To address this problem, we presented a benchmark generating tool that is able to generate time-varying networks with evolving community structure. Starting out from the basic behavior that evolving communities may have, mainly growing, shrinking, splitting or merging with other communities, the method generates time-varying networks where the number of communities and its behavior across time can be tuned by the user with great detail. This benchmarking tool will be very useful in evaluating current and new evolving community detection algorithms.

Another subject that has been studied and presented in this document is the analysis of dynamic processes on multiplex networks. Multiplex networks are a particular case of multilayer networks, in which the same set of nodes is used in all layers, with a one to one correspondence, while the connectivity patterns between nodes may differ across layers. The most interesting feature about multiplex networks is that the dynamical effects that we observe when we consider multiple layers of connectivity cannot be observed when analyzing the aggregation of all the layers in a single network. Here, we have presented the study of the interplay between two epidemic spreading processes, both of them taking place in the same population, but transmitted by different mechanisms. Inspired by

how humans are responsive to the alerts spread through their network of contacts, we have studied how the spreading of information affects the spreading of an infectious disease. Indeed, if individuals are informed about the presence of an epidemics, they can take measures to prevent infection, thus reducing the final impact of the disease in the population. To quantify this effect, we designed a model that considered an information spreading process through a network of social contacts and an epidemic spreading process running in a network with different connectivity, accounting for interactions that require physical contact or proximity. The interaction between the two processes follows the intuition that aware individuals reduce their chances to get infected and infected individuals are active transmitters of information. On this setup, we were interested in studying how does the information spreading process affect the onset of the epidemic spreading process. Very interestingly, we found that the onset of the epidemics does not depend on the awareness until a certain critical value of the infectivity rate of the information process —what we call the meta-critical point—, and after that, the dependence is clearly non-linear. Inspired by the awareness campaigns promoted through mass media, we also studied the effect of a massive broadcast of awareness, by means of an additional entity that regularly transmitted information over the awareness network. We found that the effect of mass media is very relevant, being able to shift the onset of the epidemics and eliminating the meta-critical point.

In this document we have also shown the convenience of approaching an experimental neuroscience problem using the complex networks approach. One of the still open challenges in biology and neuroscience is to understand the structure and function of neuronal systems. This ambitious problem is nowadays approached from many different perspectives, where one of the major approaches is to study *in vitro* neuronal structures. Indeed, neurons collected from a subject in an embryonic state can be dissociated and cultured in petri dishes, where neurons are able to self-organize until, after several days, they reach maturity. In this phase, if the mobility of the neurons has not been impaired, they form clusters of neurons that connect to one another to create a *de novo* neuronal network with rich spontaneous activity patterns. The study of the new structure formed is challenging, given that many connections are established between clusters but not all of them are operative, thus posing a distance between the physical observation and the real functionality of the culture. In the work presented, we approached the study of the functional network of several clustered neuronal cultures, and characterized the associated complex networks. To do so, we used data

obtained by monitoring the self-sustained activity of the cultures, and proposed a method to statistically infer this connectivity. We followed the simple principle that if two clusters activate with small time difference, then probably the first cluster to fire caused the activation of the second. Proceeding in this manner for the whole firing sequence recorded, we were able to construct the directed functional networks of the clustered cultures. The characterization of the functional networks associated to the signal recorded revealed a very interesting fact: clustered cultures exhibit positive assortativity values and show a rich-club structure, meaning that there is a preference for clusters to link to other clusters that share similar functional connectivity. This observation suggests that the process of establishing connections during neuronal growth follows an intrinsic survival mechanism.

## FUTURE PERSPECTIVES

During the development of this thesis, I have had the opportunity to explore multiple topics in complex networks. I have realized how immense this interdisciplinary field is, and perhaps due to this interdisciplinarity, how fast is it growing. During these years I have witnessed the field's growing interest towards multilayer networks as an appropriate representation of systems containing different types of interactions. This has caused the explosive development of new methods aimed to represent and characterize such networks, as well as the adaption and study of all sort of dynamical processes on top of these structures. I believe that the multilayer fever has come to stay, and that new high impact research will not neglect this newly considered multidimensionality.

Another impression derived from my years of study is that for a long time, network science has had the aim of characterizing structures and dynamics. The goal, fairly enough, has always been to understand the underlying organizational principles behind the complex systems at study. However, I believe that in the future years a transition is to be expected. In the same way as the quest of power-laws in real world systems suffered a decrease of attention several years ago, I believe that purely descriptive methods will share the same fate. Instead, I believe that the future of network science requires stepping towards the *prediction* of the behavior of complex systems.

How to encompass the design of robust predictive methods with the need of taking into account the multidimensional character of a system is something that

I have yet to discover. My journey towards gaining this knowledge will start by learning inference techniques applied to networks, to be able to design robust, generic and statistically grounded methods. A natural starting point towards this direction would be to broaden my previous knowledge on the modular structure and dynamics on and of networks with the new developments on network inference, and in particular in modular time-varying settings.

I would like to design generic methods, but I am specially interested in applying them to social data. Nowadays, the increasing integration of technology into our lives has created unprecedented volumes of data on society's everyday behavior. Such data opens up exciting new opportunities to work towards a quantitative understanding of our complex social systems. However, the captured information may lack reliability, something unacceptable if we are willing to use this information as a proxy of real social behavior. Moreover, social behavior, and in turn, online social data, evolves in time, which makes the representation by means of a time-varying network is not only convenient but indispensable. Therefore, I would like to be involved in the creation of a set of inference-based methods focused to assess the validity of the evolving structure of social networks. Aside from validation tasks, these methods would also be able to infer the future evolution of a network based on the previous temporal information, to correct such data, and to detect and predict critical events.

# A

# APPENDIX

## A.1 EXPERIMENTAL SETUP OF THE NEURONAL CULTURES AND DATA TREATMENT

All the following experiments used cortical neurons from rat embryonic brains. All procedures were approved by the Ethical Committee for Animal Experimentation of the University of Barcelona, under order DMAH-5461.

### CLUSTERED NEURONAL CULTURES

In all the experiments we used cortical neurons from $18-19$ day old Sprague–Dawley rat embryos. Following standard procedures [191, 165] dissection was carried out in ice–cold L–15 medium enriched with 0.6% glucose and gentamycin (Sigma-Aldrich). Cortices where gently extracted and dissociated by repeated pipetting.

Cortical neurons were plated onto 13 mm glass coverslips (Marienfeld-Superior) that incorporated a poly-dimethylsiloxane (PDMS) mold. The PDMS restricted neuronal growth to isolated, circular cavities 3 mm in diameter. Prior plating, glasses where washed in 70% nitric acid for 2 h, rinsed with double–distilled water (DDW), sonicated in ethanol and flamed. In parallel to glass cleaning, and following the procedure described by Orlandi *et al.* [165], several 13 mm diameter layers of PDMS $1-2$ mm thick were prepared and subsequently pierced with 3 mm diameter biopsy punchers (Integra-Miltex). Each pierced PDMS mold typically contained 4 to 6 cavities. The PDMS molds were then attached to the glasses and the combined structure autoclaved at 120°C, firmly adhering to one

another. For each dissection we prepared 12 identical glass-PDMS structures, giving rise to about 80 cultures of 3 mm in diameter. Neurons were plated in the PDMS cavities with a nominal density of 500 neurons/mm$^2$, and incubated in plating medium at 37°C, 5% $CO_2$ and 95% humidity. Plating medium consisted in 5% of foetal calf serum (FCS, Invitrogen), 5% of horse serum (HS, Inivtrogen), and 0.1% B27 (Sigma) in MEM Eagle's-L-glutamate (Invitrogen). MEM was enriched with gentamicin (Sigma), the neuronal activity promoter Glutamax (Sigma) and glucose.

Upon plating, the lack of adhesive proteins in the glass substrate rapidly favored cell–cell attachment and, gradually, the formation of islands of highly compact neuronal assemblies or *clusters* that minimized the surface contact with the substrate. Clustered cultures formed quickly. By *day in vitro* (DIV) 2 the culture encompasses dozens of small aggregates that coalesce and grow in size as the culture matures. Spontaneous activity and connections between clusters were observed by DIV 5. Clusters at this stage of development also anchored at the surface of the glass and, although they continued growing and developing connections, their number and position remained stable along the next 2 weeks. At the moment of measuring, each PDMS cavity contained an independent culture formed by $20 - 40$ interconnected clusters.

Clustered cultures were maintained for about 3 weeks, as follows. At DIV 3 the plating medium was refreshed. This promoted glial cells to develop, ensuring the survival of the culture. At DIV 5 the medium was switched from plating to *changing medium* (containing 0.5% FUDR, 0.5% Uridine, and 10% HS in enriched MEM) to limit glial cell division. Three days later, the medium was replaced to *final medium* (enriched MEM with 10% HS), which was then refreshed periodically every three days.

### HOMOGENEOUS NEURONAL CULTURES

Overnight exposure of the glass coverslips to poly-l-lisine (PLL, Sigma) provided a layer of adhesive proteins for the neurons to quickly anchor upon plating, leading to cultures with a homogeneous distribution of neurons over the substrate. The remaining steps in the preparation and maintenance of the cultures were identical as the clustered ones, i.e. we used the same nominal neuronal density for plating, we included PDMS pierced molds to confine neuronal growth in cavities 3 mm in diameter, and we refreshed the culture mediums in the same manner.

## EXPERIMENTAL SETUP AND PROCEDURE

### *Standard experiments*

To measure the spontaneous activity in the clustered networks we used cultures at day *in vitro* (DIV) $5-16$, i.e. covering about two weeks of development. Cultures started to degrade by DIV 25, and therefore we did not use cultures older than 3 weeks in our experiments.

Activity in neuronal cultures was monitored through fluorescence calcium imaging [207, 97], which allows the detection of neuronal activity by the biding of $Ca^{+2}$ ions to a fluorescence probe upon firing. Prior to recording, the culture under study was incubated for 40 min in External Medium (EM, consisting of 128 mM NaCl, 1 mM $CaCl_2$, 1 mM $MgCl_2$, 45 mM sucrose, 10 mM glucose, and 0.01 M Hepes; pH 7.4) in the presence of Fluo-4-AM (Invtrogen). We used 4 $\mu$l Fluo4 in a volume of 2 ml EM. After incubation, the culture was washed with fresh EM and placed in the observation chamber, consisting of a standard glass bottom culture dish, filled with 4 ml EM, and with its wall and cover screened from external light. To minimize accidental damage to the aggregates during the manipulation of the cultures, the PDMS pierced mold was left in contact with the glass during both incubation and the actual experiment.

The observation chamber was mounted on Zeiss Axiovert inverted microscope equipped with a high–speed CMOS camera (Hamamatsu Orca Flash 2.8). We used an objective of 2.5X combined with a 0.32X optical zoom. These settings provided a final field of view of $7.6 \times 3.4$ (width $\times$ height) $mm^2$ that supported the recording of 1 or 2 PDMS cavities simultaneously.

The fluorescence signal of the clusters' spontaneous activity was recorded with the software Hokawo 2.5, provided by the camera vendor. We used acquisition speeds in the range $83-100$ frames per second (fps), corresponding to, respectively, a time interval of $12-10$ ms between consecutive frames. These acquisition speeds were selected to optimize the balance between image quality, sufficient time resolution, and minimum light intensity. The latter was particularly important to minimize photo-damage and photo-bleaching, and allowed neuronal cultures to be studied with optimal conditions for at least 3 h. Individual frames were acquired as 8–bit grey–scale images, a size of $940 \times 400$ pixels, and a spatial resolution of 8.51 $\mu$m/pixel. A typical recording lasted for $30-60$ min. The frequency of firing strongly varied from cluster to cluster, and ranged from 0.2

firings/min for the least active clusters to 2 firings/min for the highest active ones. Experiments were carried out at room temperature.

Measurements in homogeneous cultures were carried out in the same way, with the only difference that the recording speed was increased to 100 – 150 fps to take into account the fast propagation of activity fronts in these preparations, as observed for instance in the study of Orlandi *et al.* [165].

*Resilience experiments*

We considered two groups of resilience experiments. In a first group, we monitored the gradual degradation of network activity due to photo-damage. In a second group, we measured the decay in activity as a consequence of the gradual weakening of the excitatory connectivity.

In the first group of measurements, we first considered a clustered culture and measured activity uninterruptedly along 2 hours, with neurons continuously exposed to strong light. We then divided the sequence in blocks of 30 min, and determined for each block the average network activity by counting the number of bursting episodes within the block. Next, we switched to a homogeneous culture from the same batch (i.e. identical nominal density and age) and carried out the same protocol. We finally analyzed, for each kind of network, the decay in network activity as a function of time. Data was averaged over at least 3 different pairs of cultures to take into account network variability. The comparison between the two networks indicated which topology exhibited higher resistance to degradation in neuronal activity.

In the second group of measurements, we compared the change in activity between a clustered and a homogeneous culture during gradual weakening of neuronal connectivity. The weakening was achieved by progressive application of CNQX (see 'Pharmacology'), an AMPA-glutamate receptor antagonist in excitatory neurons. We first measured the clustered network and thereafter the homogeneous one. In both cases., starting at [CNQX] = 0 nM, we recorded activity for 15 min, then increased the concentration of the drug and measured again. We repeated the procedure until activity ceased. The corresponding CNQX value hinted at the robustness of the network to a global failure of its connectivity.

## PHARMACOLOGY

The pharmacological protocols described below were used identically in clustered and homogeneous cultures.

### *Inhibitory connections*

The *in vitro* networks contain both excitatory and inhibitory connections. However, for sake of simplicity in the comparison between experiments and model, in the experiments at DIV 6 and above we completely blocked $\gamma$-aminobutyric acid (GABA) inhibitory synapses with 40 $\mu$M of the antagonist bicuculine methiodide (Sigma). The drug was applied 5 min before the actual recordings for the drug to take effect.

Hence, spontaneous activity in our experiments is solely driven by excitatory connections. Although the balance between excitation and inhibition shapes the major traits of spontaneous activity, such as the average firing rate of the network, we verified that the presence of inhibition did not qualitatively modify the results presented here.

We left active inhibitory synapses for experiments at DIV 5 since at this early stages of development GABA has a depolarizing effect and therefore an excitatory action [83, 198]. Its blockade would effectively reduce excitation and silence the network.

### *Network connectivity weakening through CNQX*

In the studies of network resilience to the weakening of connectivity, we studied the decay in spontaneous activity as a result of the gradual application of 6-cyano-7-nitroquinoxaline-2,3-dione (CNQX, Sigma), an AMPA-glutamate receptor antagonists in excitatory neurons. For [CNQX] $= 0$ the connectivity strength between neurons is maximum. As [CNQX] is administered, the efficacy of excitatory connections steadily diminishes, which is accompanied by a reduction in spontaneous activity. High CNQX concentrations lead to a complete halt in activity. In the measurements we used CNQX concentrations in the range $0 - 2000$ nM, in quasi-logarithmic steps. We left the culture unperturbed for 5 min after each CNQX application for the drug to reach steady–state effects.

## DATA ANALYSIS FOR CLUSTERED CULTURES

The acquired images (recorded at a typical speed of 100 fps) were first analyzed with the Hokawo 2.5 software to extract the fluorescence intensity of each cluster as a function of time. The regions of interest (ROIs) were chosen manually and typically covered an area of $40 \times 40$ pixels, each ROI corresponding to a single cluster. As illustrated in Fig. 4.2 and Figure 4.4, activity is characterized by a stable baseline (resting state) interrupted by peaks of fluorescence that correspond to neuronal firing. At the onset of firing, the fluorescence signal increases abruptly due to the fast intake of $Ca^{2+}$ ions. Fluorescence then reaches a maximum, and slowly decays back to the baseline in $2-5$ s.

The algorithm that we used to detect the onset of firing for each cluster was as follows. We first corrected the fluorescence signal $\tilde{F}(t)$ from small drifts, and calculated the resting fluorescence level $F_0$ by discarding the data points with an amplitude two times above the standard deviation (SD) of the signal. The corrected signal was then expressed as $F(t) \equiv \Delta\tilde{F}/F_0 = (\tilde{F} - F_0)/F_0$. We next took $F(t)$ and computed its derivative $\dot{F}(t)$ in order to detect fast changes in the fluorescence signal. Finally, the beginning of a burst in the data series was defined as the time where a maximum in $\dot{F}(t)$ was accompanied by values of $F(t)$ two times above the SD of the background signal, and for at least 5 frames.

### *Reliability in detecting the clusters' ignition times*

Three major tests were carried out to assess the reliability of our analysis. In a first one, we measured spontaneous activity at 200 fps, i.e. twice the standard recording speed, but used stronger light to compensate for the lower exposure time. We next analyzed the data, re-sampled the image sequence down to 100 fps and compared the results with the original acquisition. We observed that the detection of the onset times improved only by about 15%, which did not justify the excess of light and the associated damage to the neurons. In a second test, we measured spontaneous activity in a culture using identical light settings but considering different acquisition rates, namely 100, 150, and 200 fps. We then selected ignition sequences that were as similar as possible in all three measurements, and compared the results. We observed that only in the few cases where the clusters fired with strong amplitudes the increased speed enhanced detection,

and again by 15%. For the rest of the cases, the higher speeds actually worsened the analysis due to the poorer signal-to-noise ratio.

Finally, in a third test, we used sub-frame resolution analysis tools to evaluate the importance of finer ignition time values. Following Orlandi *et al.* [165], we considered the approach of fitting two straight lines at the vicinity of each initially detected firing. A first fit included the 100 points of the background signal that preceded ignition, and a second one extended to the 10 points that correspond to the fast rise in fluorescence. The crossing value of the two lines provided an onset time that refined the initially measured value. The better accuracy effectively increased the discrimination of sequences that were initially identified as simultaneous events. However, since these events are rare (by 5%), the finer temporal resolution had practically no effect in the construction of the functional networks and the derived analysis.

*Activity propagation times and burst duration*

The time delay $t_p$ in the propagation of activity between neighboring clusters was measured in control experiments with high acquisition rates. We concluded that $t_p$ varied in the range $10 \leq t_p \leq 100$ ms, with an average value $\bar{t}_p = 50$ ms. Other studies in clustered networks provided similar results [208]. With the detection algorithm described above and standard experiments at $83 - 100$ fps, we could appraise the activation sequence in 93% of the cases. The remaining 7% corresponded to clusters that ignited in the same frame or time bin, and were treated as simultaneous events.

Given the propagation time $t_p$, we observed that the total duration of a bursting episode, i.e. from the first occurrence of firing in a group o clusters to the last one, had to depend on the number of participating clusters. Hence we could not provide a unique characteristic window for the duration of a burst.

## DATA ANALYSIS FOR HOMOGENEOUS CULTURES

Recordings in homogeneous cultures provide the activity of $\simeq 2000$ neurons in an circular area 3 mm in diameter. Neurons are marked individually as regions of interest in the images and the corresponding fluorescence time traces extracted using custom–made software. Ignition times for each neuron were next obtained

by using the sub–frame resolution method described above (detailed in Ref. [165]), and that consisted in fitting two straight lines to the fluorescence data, a first fit encompassing the 100 points in the background region prior to firing, and a second fit including the 10 points during the fast rise in fluorescence that follows ignition. The crossing point of the two lines provided the onset of firing.

The construction of the directed functional networks for the homogeneous cultures was carried out identically as the clustered ones.

### ADDITIONAL CONTROL EXPERIMENTS

Recordings in clustered cultures typically lasted for 1 h and contained between $\simeq$ 50 bursts in the quietest networks and $\simeq$ 450 bursts in the most active ones. To test whether 50 bursts sufficed to draw the functional networks, we carried out a control experiment in which we monitored spontaneous activity along 2 h in a standard clustered culture, measured at DIV 12 and containing 42 nodes (Figure S6). We then analyzed the data using two different procedures. In the first one we drew the functional connectivity using the data extracted from the entire recording, and determined its assortativity values. In the second procedure, we separated the recorded sequence in three blocks, each 40 min long, built the functional connectivity for each block, and computed the respective assortative values. The studied culture fired in a sustained manner at a rate of 1.12 bursts/min, and procured a total of 134 bursts. Thus, each block typically contained about 45 bursts.

The results (shown in Figs. A.1 and A.2) led to two major conclusions. First, that the functional connectivity is very similar among the blocks, and between any of the blocks and the entire recording, providing assortativity values that are compatible within statistical error. And second, that the first 40 min of recording (with 45 bursts only) sufficed to shape the major traits of the functional network, therefore validating our strategy of using 1h of acquisition to procure a reliable estimate of the functional connectivity of the network and its assortative traits.

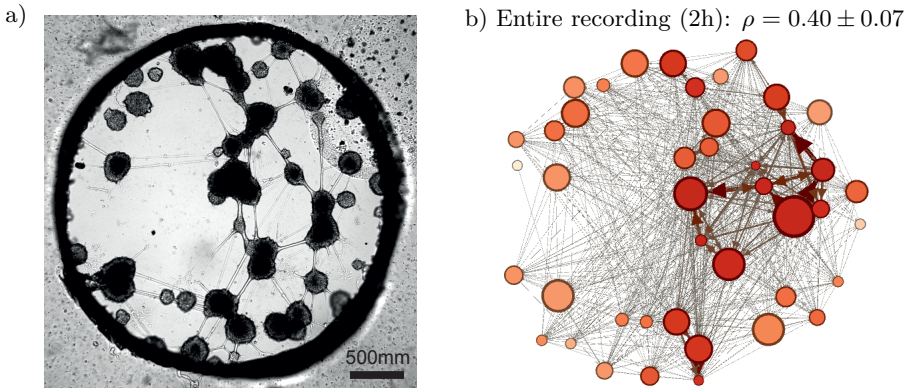



Figure A.1: Control experiment. **a)** Bright field image of a clustered network whose spontaneous activity has been recorded for 2 h. The average bursting rate of the network is 1.12 bursts/min. **b)** Corresponding functional network. The size of the nodes is proportional to the size of the actual clusters, and their color is proportional to their strength. The weights of the links are both color and thickness coded. The darker the color, the higher the value of the observable.

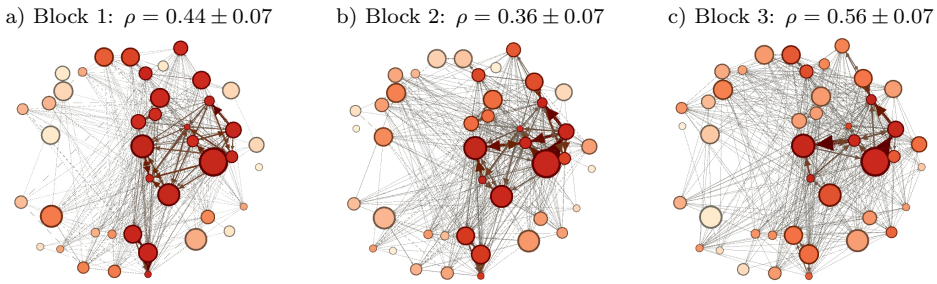



Figure A.2: Functional networks for the control experiment, where the analysis of the 2h recording is done in three blocks of 40 min in duration each, and containing 45 bursts. The blocks show very similar traits between them, as well as with the entire recording. The blocks exhibit similar assortativity values, and share both the most important links and nodes' strengths. $\rho$ indicates the assortativity value of the depicted network, averaged over the Pearson and Spearman formulations.

## A.2 ALTERNATIVE METHOD FOR CONSTRUCTING THE FUNCTIONAL NETWORK OF FIRING NEURONS

Mutual information [84, 193] is a particular case of the Kullback-Leibler divergence [124], an information-theoretic measure of the distance between two probability distributions. In fact, the mutual information between two stochastic variables $X$ and $Y$ provides an estimation of the amount of information gained about $X$ when $Y$ is known.

Let us indicate by $\{s_\ell^{(i)}\}$ the time series corresponding to the $i$-th cluster, with $\ell = 1, 2, ..., L$ and $L$ the total number of time frames involved in the observation process. The time series adopted for the successive analysis are obtained by mapping the observed train of cluster activations to another time series termed *walk*, defined by

$$x_\ell^{(i)} = \sum_{l=1}^{\ell} \left[ s_l^{(i)} - \langle s_\ell^{(i)} \rangle \right]. \tag{A.1}$$

In the specific case of our analysis, the mutual information between two time series $\{x_\ell^{(i)}\}$ and $\{x_\ell^{(j)}\}$, corresponding to two different clusters, is interpreted as the amount of correlation between the dynamics of cluster $i$ and $j$. In general, the time scale of the correlation between two time series is not known *a priori*. Such a time scale corresponds to the time delay required to maximize the gain of information. Therefore, in the spirit of Fraser and Swinney [79], we define the time delayed mutual cross information between $\{x_\ell^{(i)}\}$ and $\{x_\ell^{(j)}\}$ by

$$I(x^{(i)}, x^{(j)}; \tau) = -\sum_{\mu,\nu} p_{\mu\nu}^{(i,j)}(\tau) \log \frac{p_{\mu\nu}^{(i,j)}(\tau)}{p_\mu^{(i)} p_\nu^{(j)}}, \tag{A.2}$$

where $\mu$ and $\nu$ are indices running over some partition of the observed time series. In Eq. (A.2), $p_\mu^{(i)}$ indicates the probability to find a value of time series $\{x_\ell^{(i)}\}$ in the $\mu$-th interval, $p_\nu^{(j)}$ is the probability to find a value of time series $\{x_\ell^{(j)}\}$ in the $\nu$-th interval, whereas $p_{\mu\nu}^{(i,j)}$ denotes the joint probability to observe a firing from the $i$-th cluster falling in the $\mu$-th interval and a firing from the $j$-th cluster falling in the $\nu$-th interval exactly $\tau$ time frames later.

For the sake of simplicity, in the following we will adopt the more concise notation $I_{ij}(\tau) = I(x^{(i)}, x^{(j)}; \tau)$ to indicate the time delayed mutual cross information. Finally, in order to gain the highest amount of information about the dynamics of cluster $i$ by observing cluster $j$, we consider only the maximum value $I_{ij}^{\max} = \max_\tau [I_{ij}(\tau)]$ of $I_{ij}(\tau)$ with respect to the time delay $\tau$.

We estimate the importance of the observed amount of correlation by performing the above analysis on surrogate data. Surrogates adopted in this study are time series generated by randomly reshuffling the temporal observations of the firing series $\{s_\ell^{(i)}\}$, for each cluster separately. Such a procedure destroys any correlation between pairs of time series while preserving the empirical probability distribution, thus allowing to test the null hypothesis that the observed correlation is obtained by chance.

We indicate by $\{\tilde{x}_\ell^{(i)}\}$ the walk corresponding to the surrogate obtained from time series $\{x_\ell^{(i)}\}$ and with $\tilde{I}_{ij}(\tau)$ the time delayed mutual cross information between $\{\tilde{x}_\ell^{(i)}\}$ and $\{\tilde{x}_\ell^{(j)}\}$. We perform 200 independent random realizations of surrogates for each pair $(i, j)$ and we estimate the corresponding expected value $\langle \tilde{I}_{ij}^{\max} \rangle$ of the maximum mutual cross–information, as well as the root mean square $\tilde{\sigma}_{ij}$ of the underlying distribution.

Hence, we fix *a priori* the significance $\alpha$ of the hypothesis testing and we estimate the $z$-score corresponding to each pair $(i, j)$ by $z_{ij} = (I_{ij}^{\max} - \langle \tilde{I}_{ij}^{\max} \rangle)/\tilde{\sigma}_{ij}$. Therefore, the observed correlation between cluster $i$ and $j$ is said to be statistically significant if $1 - \mathrm{erf}(z_{ij}/\sqrt{2}) \leq \alpha$, where erf is the standard error function. Finally, we obtain the functional network of clusters by building the weight matrix $\mathbf{W}$ whose elements are defined by $w_{ij} = z_{ij}$ if $1 - \mathrm{erf}(z_{ij}/\sqrt{2}) \leq \alpha$, and $w_{ij} = 0$ if $1 - \mathrm{erf}(z_{ij}/\sqrt{2}) > \alpha$.

UNIVERSITAT ROVIRA I VIRGILI
FROM COMMUNITY STRUCTURE TO THE PHYSICS OF MULTIPLEX NETWORKS
Clara Granell Martorell

# BIBLIOGRAPHY

[1] Achard, S., Salvador, R., Whitcher, B., Suckling, J., and Bullmore, E. A resilient, low-frequency, small-world human brain functional network with highly connected association cortical hubs. *The Journal of Neuroscience 26*, 1 (2006), 63–72.

[2] Aguirre, J., Papo, D., and Buldu, J. M. Successful strategies for competing networks. *Nat Phys 9*, 4 (04 2013), 230–234.

[3] Alba, R. D. A graph-theoretic definition of a sociometric clique? *Journal of Mathematical Sociology 3*, 1 (1973), 113–126.

[4] Albert, R., and Barabási, A.-L. Statistical mechanics of complex networks. *Reviews of modern physics 74* (2002), 47.

[5] Albert, R., Jeong, H., and Barabasi, A.-L. Error and attack tolerance of complex networks. *Nature 406*, 6794 (07 2000), 378–382.

[6] Anderson, R. M., and May, R. M. Coevolution of hosts and parasites. *Parasitology 85* (10 1982), 411–426.

[7] Anderson, R. M., May, R. M., and Anderson, B. *Infectious diseases of humans: dynamics and control*, vol. 28. Wiley Online Library, 1992.

[8] Andersson, H., and Britton, T. *Stochastic epidemic models and their statistical analysis*, vol. 151. Springer Science & Business Media, 2012.

[9] Arenas, A., Duch, J., Fernández, A., and Gómez, S. Size reduction of complex networks preserving modularity. *New Journal of Physics 9* (2007), 176.

[10] Arenas, A., Fernández, A., and Gómez, S. Analysis of the structure of complex networks at different resolution levels. *New Journal of Physics 10*, 053039 (2008).

[11] Avalos-Gaytán, V., Almendral, J. A., Papo, D., Schaeffer, S. E., and Boccaletti, S. Assortative and modular networks are shaped by adaptive synchronization processes. *Phys. Rev. E 86* (Jul 2012), 015101.

Bibliography

[12] Axelrod, R., and Hamilton, W. D. The evolution of cooperation. *Science 211*, 4489 (1981), 1390–1396.

[13] Barabási, A.-L., and Albert, R. Emergence of scaling in random networks. *Science 286*, 5439 (1999), 509–512.

[14] Barrat, A., Fernandez, B., Lin, K. K., and Young, L.-S. Modeling temporal networks using random itineraries. *Phys. Rev. Lett. 110* (Apr 2013), 158702.

[15] Barrett, L., Henzi, S. P., and Lusseau, D. Taking sociality seriously: the structure of multi-dimensional social networks as a source of information for individuals. *Philosophical Transactions of the Royal Society of London B: Biological Sciences 367*, 1599 (2012), 2108–2118.

[16] Bassett, D. S., Porter, M. A., Wymbs, N. F., Grafton, S. T., Carlson, J. M., and Mucha, P. J. Robust detection of dynamic community structure in networks. *Chaos: An Interdisciplinary Journal of Nonlinear Science 23*, 1 (2013), 013142.

[17] Bassett, D. S., Wymbs, N. F., Porter, M. A., Mucha, P. J., Carlson, J. M., and Grafton, S. T. Dynamic reconfiguration of human brain networks during learning. *Proceedings of the National Academy of Sciences 108*, 18 (2011), 7641–7646.

[18] Battiston, F., Nicosia, V., and Latora, V. Structural measures for multiplex networks. *Phys. Rev. E 89* (Mar 2014), 032804.

[19] Baxter, G. J., Dorogovtsev, S. N., Goltsev, A. V., and Mendes, J. F. F. Avalanche collapse of interdependent networks. *Phys. Rev. Lett. 109* (Dec 2012), 248701.

[20] Berlingerio, M., Coscia, M., Giannotti, F., Monreale, A., and Pedreschi, D. Foundations of multidimensional network analysis. In *Advances in Social Networks Analysis and Mining (ASONAM), 2011 International Conference on* (July 2011), pp. 485–489.

[21] Bettencourt, L. M. A., Stephens, G. J., Ham, M. I., and Gross, G. W. Functional structure of cortical neuronal networks grown *in vitro*. *Phys. Rev. E 75* (Feb 2007), 021915.

[22] Bjørnstad, O. N., Finkenstädt, B. F., and Grenfell, B. T. Dynamics of measles epidemics: estimating scaling of transmission rates using a time series sir model. *Ecological Monographs 72*, 2 (2002), 169–184.

[23] Blondel, V. D., Guillaume, J.-L., Lambiotte, R., and Lefebvre, E. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment 2008*, 10 (2008), P10008.

[24] Boccaletti, S., Bianconi, G., Criado, R., Del Genio, C. I., Gómez-Gardeñes, J., Romance, M., Sendina-Nadal, I., Wang, Z., and Zanin, M. The structure and dynamics of multilayer networks. *Physics Reports 544*, 1 (2014), 1–122.

[25] Boccaletti, S., Latora, V., Moreno, Y., Chavez, M., and Hwang, D.-U. Complex networks : Structure and dynamics. *Phys. Rep. 424*, 4-5 (2006), 175–308.

[26] Boettcher, S., and Percus, A. G. Optimization with extremal dynamics. *complexity 8*, 2 (2002), 57–62.

[27] Boguñá, M., Castellano, C., and Pastor-Satorras, R. Langevin approach for the dynamics of the contact process on annealed scale-free networks. *Phys Rev E Stat Nonlin Soft Matter Phys 79*, 3 Pt 2 (2009), 036110.

[28] Boguñá, M., and Pastor-Satorras, R. Epidemic spreading in correlated complex networks. *Phys. Rev. E 66* (Oct 2002), 047104.

[29] Bollobas, B. *Modern Graph Theory*. Springer, 1998.

[30] Bonifazi, P., Goldin, M., Picardo, M. A., Jorquera, I., Cattani, A., Bianconi, G., Represa, A., Ben-Ari, Y., and Cossart, R. Gabaergic hub neurons orchestrate synchrony in developing hippocampal networks. *Science 326*, 5958 (2009), 1419–1424.

[31] Brauer, F., Castillo-Chavez, C., and Castillo-Chavez, C. *Mathematical models in population biology and epidemiology*, vol. 1. Springer, 2001.

[32] Bródka, P., Kazienko, P., MusialĆ, K., and Skibicki, K. Analysis of neighbourhoods in multi-layered dynamic social networks. *International Journal of Computational Intelligence Systems 5*, 3 (2012), 582–596.

[33] Bródka, P., Saganowski, S., and Kazienko, P. Ged: the method for group evolution discovery in social networks. *Social Network Analysis and Mining 3*, 1 (2013), 1–14.

[34] Brummitt, C. D., Lee, K.-M., and Goh, K.-I. Multiplexity-facilitated

cascades in networks. *Phys. Rev. E 85* (Apr 2012), 045102.

[35] Buldyrev, S. V., Parshani, R., Paul, G., Stanley, H. E., and Havlin, S. Catastrophic cascade of failures in interdependent networks. *Nature 464*, 7291 (04 2010), 1025–1028.

[36] Bullmore, E., and Sporns, O. Complex brain networks: Graph theoretical analysis of structural and functional systems. *Nature Reviews Neuroscience 10*, 3 (2009), 186–198. cited By (since 1996)963.

[37] Byungjoon Min, K.-I. G. Layer-crossing overhead and information spreading in multiplex social networks. *arXiv preprint arXiv:1307.2967* (2013).

[38] Chakrabarti, D., Kumar, R., and Tomkins, A. Evolutionary clustering. In *KDD '06: Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining* (New York, NY, USA, 2006), ACM, pp. 554–560.

[39] Chen, W., and D'Souza, R. M. Explosive percolation with multiple giant components. *Phys. Rev. Lett. 106* (Mar 2011), 115701.

[40] Chen, W., Zheng, Z., Jiang, X., and D'Souza, R. M. Multiple discontinuous percolation transitions on scale-free networks. *Journal of Statistical Mechanics: Theory and Experiment 2015*, 4 (2015), P04011.

[41] Clauset, A., Shalizi, C. R., and Newman, M. E. Power-law distributions in empirical data. *SIAM review 51*, 4 (2009), 661–703.

[42] Cohen, R., Erez, K., ben Avraham, D., and Havlin, S. Breakdown of the internet under intentional attack. *Phys. Rev. Lett. 86* (Apr 2001), 3682–3685.

[43] Cohen, R., and Havlin, S. *Complex networks: structure, robustness and function*. Cambridge University Press, 2010.

[44] Colizza, V., Pastor-Satorras, R., and Vespignani, A. Reaction–diffusion processes and metapopulation models in heterogeneous networks. *Nat Phys 3* (Jan. 2007), 276–282.

[45] Cox, D. Renewal theory. *Science Paperback Edition* (1967).

[46] Cozzo, E., Baños, R. A., Meloni, S., and Moreno, Y. Contact-based social contagion in multiplex networks. *Phys. Rev. E 88* (Nov 2013), 050801.

[47] Cozzo, E., Kivelä, M., Domenico, M. D., Solé-Ribalta, A., Arenas, A., Gómez, S., Porter, M. A., and Moreno, Y. Structure of triadic relations in multiplex networks. *New Journal of Physics 17*, 7 (2015), 073029.

[48] Criado, R., Flores, J., del Amo, A. G., Gómez-Gardeñes, J., and Romance, M. A mathematical model for networks with structures in the mesoscale. *International Journal of Computer Mathematics 89*, 3 (2012), 291–309.

[49] Darabi Sahneh, F., and Scoglio, C. Competitive epidemic spreading over arbitrary multilayer networks. *Phys. Rev. E 89* (Jun 2014), 062817.

[50] Darst, R. K. `http://rkd.zgib.net/proj/multiplex/`.

[51] De Domenico, M., Lancichinetti, A., Arenas, A., and Rosvall, M. Identifying modular flows on multilayer networks reveals highly overlapping organization in interconnected systems. *Phys. Rev. X 5* (Mar 2015), 011027.

[52] De Domenico, M., Solé-Ribalta, A., Cozzo, E., Kivelä, M., Moreno, Y., Porter, M. A., Gómez, S., and Arenas, A. Mathematical formulation of multilayer networks. *Phys. Rev. X 3* (Dec 2013), 041022.

[53] De Domenico, M., Sole-Ribalta, A., Omodei, E., Gomez, S., and Arenas, A. Ranking in interconnected multilayer networks reveals versatile nodes. *Nat Commun 6* (04 2015).

[54] De Domenico, M., Solé-Ribalta, A., Gómez, S., and Arenas, A. Random walks on multiplex networks. *arXiv preprint arXiv:1306.0519* (2013).

[55] De Domenico, M., Solé-Ribalta, A., Gómez, S., and Arenas, A. Navigability of interconnected networks under random failures. *Proceedings of the National Academy of Sciences 111*, 23 (2014), 8351–8356.

[56] de Franciscis, S., Johnson, S., and Torres, J. J. Enhancing neural-network performance via assortativity. *Phys. Rev. E 83* (Mar 2011), 036114.

[57] Decelle, A., Krzakala, F., Moore, C., and Zdeborová, L. Asymptotic analysis of the stochastic block model for modular networks and its algorithmic applications. *Phys. Rev. E 84* (Dec 2011), 066106.

[58] Decelle, A., Krzakala, F., Moore, C., and Zdeborová, L. Infer-

ence and phase transitions in the detection of modules in sparse networks. *Physical Review Letters 107*, 6 (2011), 065701.

[59] Decelle, A., Krzakala, F., Moore, C., and Zdeborová, L. Inference and phase transitions in the detection of modules in sparse networks. *Phys. Rev. Lett. 107* (Aug 2011), 065701.

[60] Dickison, M., Havlin, S., and Stanley, H. E. Epidemics on interconnected networks. *Phys. Rev. E 85* (Jun 2012), 066109.

[61] Diekmann, O., Heesterbeek, H., and Britton, T. *Mathematical tools for understanding infectious disease dynamics*. Princeton University Press, 2012.

[62] Dongen, S. V. *Graph Clustering by Flow Simulation*. PhD thesis, Dutch National Research Institute for Mathematics and Computer Science, University of Utrecht, Netherlands, 2000.

[63] Dorogovtsev, S. N., Goltsev, A. V., and Mendes, J. F. F. Critical phenomena in complex networks. *Rev. Mod. Phys. 80* (Oct 2008), 1275–1335.

[64] Duch, J., and Arenas, A. Community identification using extremal optimization. *Phys. Rev. E 72*, 027104 (2005).

[65] Eckmann, J.-P., Feinerman, O., Gruendlinger, L., Moses, E., Soriano, J., and Tlusty, T. The physics of living neural networks. *Phys. Rep. 449*, 1 (Sept. 2007), 54–76.

[66] Efron, B. Computers and the theory of statistics: Thinking the unthinkable. *SIAM Review 21*, 4 (1979), pp. 460–480.

[67] Eguíluz, V. M., Chialvo, D. R., Cecchi, G. A., Baliki, M., and Apkarian, A. V. Scale-free brain functional networks. *Phys. Rev. Lett. 94* (Jan 2005), 018102.

[68] Erdös, P., and Rényi, A. On random graphs. I. *Publ. Math. Debrecen 6* (1959), 290–297.

[69] Erdös, P., and Rényi, A. On the evolution of random graphs. In *Publication of the mathematical institute of the Hungarian Academy of Sciences* (1960), pp. 17–61.

[70] Faloutsos, M., Faloutsos, P., and Faloutsos, C. On power-law relationships of the internet topology. *SIGCOMM Comput. Commun. Rev.*

*29*, 4 (Aug. 1999), 251–262.

[71] Fernández, A., and Gómez, S. Solving non-uniqueness in agglomerative hierarchical clustering using multidendrograms. *Journal of Classification 25* (2008), 43.

[72] Fiedler, M. Algebraic connectivity of graphs. *Czechoslovak mathematical journal 23*, 2 (1973), 298–305.

[73] Fisher, R. A. The use of multiple measurements in taxonomic problems. *Annals of Eugenics 7* (1936), 179.

[74] Flake, G. W., Lawrence, S., Giles, C. L., and Coetzee, F. M. Self-organization and identification of web communities. *Computer 35*, 3 (2002), 66–70.

[75] Fortunato, S. Quality functions in community detection. In *SPIE Fourth International Symposium on Fluctuations and Noise* (2007), International Society for Optics and Photonics, pp. 660108–660108.

[76] Fortunato, S. Community detection in graphs. *Phys. Rep. 486* (2010), 75.

[77] Fortunato, S. Community detection in graphs. *Physics Reports 486* (2010), 75–174.

[78] Fortunato, S., and Barthélemy, M. Resolution limit in community detection. *Proc. Natl. Acad. Sci. USA 104* (2007), 36.

[79] Fraser, A. M., and Swinney, H. L. Independent coordinates for strange attractors from mutual information. *Physical review A 33*, 2 (1986), 1134.

[80] Freeman, L. C. A set of measures of centrality based on betweenness. *Sociometry* (1977), 35–41.

[81] Freeman, L. C. Centrality in social networks conceptual clarification. *Social networks 1*, 3 (1979), 215–239.

[82] Gan, G., Ma, C., and Wu, J. *Data Clustering: Theory, Algorithms, and Applications*. Series on Statistics and Applied Probability. ASA-SIAM, 2007.

[83] Ganguly, K., Schinder, A. F., Wong, S. T., and ming Poo, M. GABA itself promotes the developmental switch of neuronal GABAergic responses from excitation to inhibition. *Cell 105*, 4 (2001), 521 – 532.

[84] Garofalo, M., Nieus, T., Massobrio, P., and Martinoia, S. Evaluation of the performance of information theory-based methods and cross-correlation to estimate the functional connectivity in cortical networks. *PLOS ONE 4*, 8 (08 2009), e6482.

[85] Girvan, M., and Newman, M. Community structure in social and biological networks. *Proc. Natl. Acad. Sci. USA 99*, 12 (2002), 7821.

[86] Gleeson, J. P. High-accuracy approximation of binary-state dynamics on networks. *Phys. Rev. Lett. 107* (Aug 2011), 068701.

[87] Glover, F. Future paths for integer programming and links to artificial intelligence. *Computers & operations research 13*, 5 (1986), 533–549.

[88] Goffman, C. And what is your erdos number? *American Mathematical Monthly* (1969), 791–791.

[89] Gómez, S., Arenas, A., Borge-Holthoefer, J., Meloni, S., and Moreno, Y. Discrete-time markov chain approach to contact-based disease spreading in complex networks. *EPL (Europhysics Letters) 89*, 3 (2010), 38009.

[90] Gómez, S., Arenas, A., Borge-Holthoefer, J., Meloni, S., and Moreno, Y. Discrete-time Markov chain approach to contact-based disease spreading in complex networks. *Europhys. Lett. 89*, 3 (Feb. 2010), 38009.

[91] Gómez, S., Díaz-Guilera, A., Gómez-Gardeñes, J., Pérez-Vicente, C. J., Moreno, Y., and Arenas, A. Diffusion dynamics on multiplex networks. *Phys. Rev. Lett. 110* (Jan 2013), 028701.

[92] Gómez, S., Gómez-Gardeñes, J., Moreno, Y., and Arenas, A. Nonperturbative heterogeneous mean-field approach to epidemic spreading in complex networks. *Physical Review E 84* (Sept. 2011), 036105.

[93] Gómez, S., Jensen, P., and Arenas, A. Analysis of community structure in networks of correlated data. *Phys. Rev. E 80* (2009), 016114.

[94] Gómez-Gardeñes, J., Gracia-Lázaro, C., Floría, L. M., and Moreno, Y. Evolutionary dynamics on interdependent populations. *Phys. Rev. E 86* (Nov 2012), 056113.

[95] Gómez-Gardeñes, J., Reinares, I., Arenas, A., and Floría, L. M. M. Evolution of cooperation in multiplex networks. *Scientific reports*

*2* (Aug. 2012).

[96] Granell, C., Gómez, S., and Arenas, A. Mesoscopic analysis of networks: Applications to exploratory analysis and data clustering. *Chaos: An Interdisciplinary Journal of Nonlinear Science 21*, 1 (2011), 016102.

[97] Grienberger, C., and Konnerth, A. Imaging calcium in neurons. *Neuron 73*, 5 (2012), 862 – 885.

[98] Gross, G. W., and Kowalski, J. M. Origins of activity patterns in self-organizing neuronal networks in vitro. *Journal of Intelligent Material Systems and Structures 10*, 7 (1999), 558–564.

[99] Guimerà, R., and Nunes Amaral, L. A. Functional cartography of complex metabolic networks. *Nature 433*, 7028 (02 2005), 895–900.

[100] Guimerà, R., and Sales-Pardo, M. Missing and spurious interactions and the reconstruction of complex networks. *Proceedings of the National Academy of Sciences of the USA 106*, 52 (2009), 22073–22078.

[101] Hagmann, P., Cammoun, L., Gigandet, X., Meuli, R., Honey, C. J., Wedeen, V. J., and Sporns, O. Mapping the structural core of human cerebral cortex. *PLOS Biol 6*, 7 (07 2008), e159.

[102] Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., and Witten, I. H. The weka data mining software: an update. *ACM SIGKDD explorations newsletter 11*, 1 (2009), 10–18.

[103] Heesterbeek, J. *Mathematical epidemiology of infectious diseases: model building, analysis and interpretation*, vol. 5. John Wiley & Sons, 2000.

[104] Hethcote, H. W. The mathematics of infectious diseases. *SIAM review 42*, 4 (2000), 599–653.

[105] Holland, P., Laskey, K. B., and Leinhardt, S. Stochastic blockmodels: Some first steps. *Soc. Netw. 5* (1983), 109–137.

[106] Holme, P. Congestion and centrality in traffic flow on complex networks. *Advances in Complex Systems 06*, 02 (2003), 163–176.

[107] Holme, P., and Saramäki, J. Temporal networks. *Physics Reports 519*, 3 (2012), 97 – 125. Temporal Networks.

[108] Hubert, L., and Arabie, P. Comparing partitions. *Journal of Classification 2*, 1 (1985), 193–218.

[109] Hwang, S., Lee, D.-S., and Kahng, B. First passage time for random walks in heterogeneous networks. *Phys. Rev. Lett. 109* (Aug 2012), 088701.

[110] Jaccard, P. The distribution of flora in the alpine zone. *The New Phytologist 11*, 2 (1912).

[111] Jaccard, P. The distribution of the flora in the alpine zone. *The New Phytologist 11* (1912), 37–50.

[112] Jain, A. K., Murty, M. N., and Flynn, P. J. Data clustering: A review. *ACM Comp. Surv. 31*, 2 (1999).

[113] Jiang, L.-L., and Perc, M. Spreading of cooperative behaviour across interdependent groups. *Scientific Reports 3* (08 2013), 2483 EP –.

[114] Jolliffe, I. *Principal component analysis*. Wiley Online Library, 2002.

[115] Jonsson, P. F., Cavanna, T., Zicha, D., and Bates, P. A. Cluster analysis of networks generated through homology: automatic identification of important protein communities involved in cancer metastasis. *BMC Bioinformatics 7* (2006), 2–2.

[116] Jutla, I. S., Jeub, L. G. S., and Mucha, P. J. A generalized louvain method for community detection implemented in matlab.

[117] Kagel, J. H., and Roth, A. E. *The handbook of experimental economics*. Princeton university press Princeton, NJ, 1995.

[118] Kaufman, L., and Rousseeuw, P. J. *Finding Groups in Data: An Introduction to Cluster Analysis*. Wiley, 2005.

[119] Keeling, M. J., and Rohani, P. *Modeling infectious diseases in humans and animals*. Princeton University Press, 2008.

[120] Kermack, W. O., and McKendrick, A. G. A contribution to the mathematical theory of epidemics. In *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences* (1927), vol. 115, The Royal Society, pp. 700–721.

[121] Kernighan, B., and Lin, S. An efficient heuristic procedure for partitioning graphs. *Bell Syst. Tech. J. 49* (1970), 291–307.

[122] Kivelä, M., Arenas, A., Barthelemy, M., Gleeson, J. P., Moreno, Y., and Porter, M. A. Multilayer networks. *Journal of Complex Networks 2*, 3 (2014), 203–271.

[123] Kovanen, L., Karsai, M., Kaski, K., Kértesz, J., and Saramäki, J. Temporal motifs in time-dependent networks. *Journal of Statistical Mechanics: Theory and Experiment 2011*, 11 (2011), P11005.

[124] Kullback, S., and Leibler, R. A. On information and sufficiency. *The Annals of Mathematical Statistics 22*, 1 (1951), 79–86.

[125] Kurant, M., and Thiran, P. Layered complex networks. *Phys. Rev. Lett. 96* (Apr 2006), 138701.

[126] Lancichinetti, A., and Fortunato, S. Limits of modularity maximization in community detection. *Phys. Rev. E 84* (Dec 2011), 066122.

[127] Lancichinetti, A., Fortunato, S., and Kertész, J. Detecting the overlapping and hierarchical community structure in complex networks. *New Journal of Physics 11*, 3 (2009), 033015.

[128] Lancichinetti, A., Fortunato, S., and Radicchi, F. Benchmark graphs for testing community detection algorithms. *Phys. Rev. E 78*, 4 (2008), 046110.

[129] Latora, V., and Marchiori, M. Efficient behavior of small-world networks. *Phys. Rev. Lett. 87* (Oct 2001), 198701.

[130] Latora, V., and Marchiori, M. Economic small-world behavior in weighted networks. *The European Physical Journal B - Condensed Matter and Complex Systems 32*, 2 (2003), 249–263.

[131] Lee, K.-M., Brummitt, C. D., and Goh, K.-I. Threshold cascades with response heterogeneity in multiplex networks. *Phys. Rev. E 90* (Dec 2014), 062816.

[132] Lee, K.-M., Kim, J. Y., kuk Cho, W., Goh, K.-I., and Kim, I.-M. Correlated multiplexity and connectivity of multiplex random networks. *New Journal of Physics 14*, 3 (2012), 033027.

[133] Lee, K.-M., Min, B., and Goh, K.-I. Towards real-world complexity: an introduction to multiplex networks. *The European Physical Journal B 88*, 2 (2015), 1–20.

[134] Leung, C., and Chau, H. Weighted assortative and disassortative networks model. *Physica A: Statistical Mechanics and its Applications 378*, 2 (2007), 591 – 602.

[135] Litvak, N., and van der Hofstad, R. Uncovering disassortativity in

large scale-free networks. *Phys. Rev. E 87* (Feb 2013), 022801.

[136] Lloyd-Smith, J. O., Schreiber, S. J., Kopp, P. E., and Getz, W. M. Superspreading and the effect of individual variation on disease emergence. *Nature 438*, 7066 (2005), 355–359.

[137] Luce, R. D. Connectivity and generalized cliques in sociometric group structure. *Psychometrika 15*, 2 (1950), 169–190.

[138] Luce, R. D., and Perry, A. D. A method of matrix analysis of group structure. *Psychometrika 14*, 2 (1949), 95–116.

[139] Massoulié, L. Community detection thresholds and the weak ramanujan property. In *Proceedings of the 46th Annual ACM Symposium on Theory of Computing* (New York, NY, USA, 2014), STOC '14, ACM, pp. 694–703.

[140] Matamalas, J. T., Poncela-Casasnovas, J., Gómez, S., and Arenas, A. Strategical incoherence regulates cooperation in social dilemmas on multiplex networks. *Scientific Reports 5* (04 2015), 9519 EP –.

[141] Meilă, M. Comparing clusterings—an information based distance. *J. Multivar. Anal. 98*, 5 (2007), 873–895.

[142] Meloni, S., Perra, N., Arenas, A., Gómez, S., Moreno, Y., and Vespignani, A. Modeling human mobility responses to the large-scale spreading of infectious diseases. *Scientific reports 1* (2011).

[143] Meunier, D., Lambiotte, R., and Bullmore, E. T. Modular and hierarchically modular organization of brain networks. *Frontiers in Neuroscience 4*, 200 (2010).

[144] Min, B., Yi, S. D., Lee, K.-M., and Goh, K.-I. Network robustness of multiplex networks with interlayer degree correlations. *Phys. Rev. E 89* (Apr 2014), 042811.

[145] Mossel, E., Neeman, J., and Sly, A. Reconstruction and estimation in the planted partition model. *Probability Theory and Related Fields 162*, 3-4 (2015), 431–461.

[146] Mucha, P. J., Richardson, T., Macon, K., Porter, M. A., and Onnela, J. P. Community Structure in Time-Dependent, Multiscale, and Multiplex Networks. *Science 328*, 5980 (May 2010), 876–878.

[147] Murray, J. Mathematical biology. *C271* (1989).

[148] Nadakuditi, R. R., and Newman, M. E. J. Graph spectra and the

detectability of community structure in networks. *Phys. Rev. Lett. 108* (May 2012), 188701.

[149] Newman, M. Assortative mixing in networks. *Phys. Rev. Lett. 89*, 20 (2002), 208701.

[150] Newman, M. Mixing patterns in networks. *Phys. Rev. E 67*, 2 (2003), 26126.

[151] Newman, M. *Networks: an introduction*. Oxford University Press, 2010.

[152] Newman, M. E. Power laws, pareto distributions and zipf's law. *Contemporary physics 46*, 5 (2005), 323–351.

[153] Newman, M. E. J. The structure of scientific collaboration networks. *Proceedings of the National Academy of Sciences 98*, 2 (2001), 404–409.

[154] Newman, M. E. J. Spread of epidemic disease on networks. *Phys. Rev. E 66* (Jul 2002), 016128.

[155] Newman, M. E. J. The structure and function of complex networks. *Society for Industrial and Applied Mathematics 45*, 2 (2003), 167–256.

[156] Newman, M. E. J. Analysis of weighted networks. *Phys. Rev. E 70* (2004), 056131.

[157] Newman, M. E. J. Fast algorithm for detecting community structure in networks. *Phys. Rev. E 69* (2004), 066133.

[158] Newman, M. E. J. Modularity and community structure in networks. *Proc. Natl. Acad. Sci. USA 103* (2006), 8577.

[159] Newman, M. E. J. Spectral methods for community detection and graph partitioning. *Phys. Rev. E 88* (Oct 2013), 042822.

[160] Newman, M. E. J., and Girvan, M. Finding and evaluating community structure in networks. *Phys. Rev. E 69*, 2 (2004), 026113.

[161] Newman, M. E. J., and Park, J. Why social networks are different from other types of networks. *Phys. Rev. E 68* (Sep 2003), 036122.

[162] Nicosia, V., and Latora, V. Measuring and modeling correlations in multiplex networks. *Phys. Rev. E 92* (Sep 2015), 032805.

[163] Noh, J. D., and Rieger, H. Random walks on complex networks. *Phys. Rev. Lett. 92* (Mar 2004), 118701.

[164] Odda, T. On properties of a well-known graph or what is your Ramsey number? *Annals N.Y. Acad. Sci. 328* (1979), 166–172.

[165] Orlandi, J. G., Soriano, J., Alvarez-Lacalle, E., Teller, S., and Casademunt, J. Noise focusing and the emergence of coherent activity in neuronal cultures. *Nature Physics 9*, 9 (2013), 582–590.

[166] Orlandi, J. G., Stetter, O., Soriano, J., Geisel, T., and Battaglia, D. Transfer entropy reconstruction and labeling of neuronal connections from simulated calcium imaging. *PLOS ONE 9*, 6 (06 2014), e98842.

[167] Palla, G., Barabási, A.-L., and Vicsek, T. Quantifying social group evolution. *Nature 446* (Apr. 2007), 664–667.

[168] Pastor-Satorras, R., Castellano, C., Van Mieghem, P., and Vespignani, A. Epidemic processes in complex networks. *Rev. Mod. Phys. 87* (Aug 2015), 925–979.

[169] Pastor-Satorras, R., and Vespignani, A. Epidemic spreading in scale-free networks. *Phys. Rev. Lett. 86* (Apr 2001), 3200–3203.

[170] Pastor-Satorras, R., and Vespignani, A. Epidemic spreading in scale-free networks. *Phys. Rev. Lett. 86* (Apr 2001), 3200–3203.

[171] Perc, M., Gómez-Gardeñes, J., Szolnoki, A., Floría, L. M., and Moreno, Y. Evolutionary dynamics of group interactions on structured populations: a review. *Journal of The Royal Society Interface 10*, 80 (2013).

[172] Perra, N., Baronchelli, A., Mocanu, D., Gonçalves, B., Pastor-Satorras, R., and Vespignani, A. Random walks and search in time-varying networks. *Phys. Rev. Lett. 109* (Dec 2012), 238701.

[173] Petri, G., and Expert, P. Temporal stability of network partitions. *Phys. Rev. E 90* (Aug 2014), 022813.

[174] Poletto, C., Meloni, S., Van Metre, A., Colizza, V., Moreno, Y., and Vespignani, A. Characterising two-pathogen competition in spatially structured environments. *Scientific Reports 5* (01 2015), 7895 EP –.

[175] Pons, P., and Latapy, M. Post-processing hierarchical community structures: Quality improvements and multi-scale view. *Theoretical Computer Science 412*, 8 (2011), 892–900.

[176] Pothen, A., Simon, H. D., and Liou, K.-P. Partitioning sparse ma-

trices with eigenvectors of graphs. *SIAM journal on matrix analysis and applications 11*, 3 (1990), 430–452.

[177] R. Guimerà, M. S.-P., and Amaral, L. Modularity from fluctuations in random graphs and complex networks. *Phys. Rev. E 70*, 2 (2004).

[178] Rand, W. M. Objective criteria for the evaluation of clustering methods. *J. Am. Stat. Assoc. 66*, 336 (1971), 846–850.

[179] Redner, S. How popular is your paper? an empirical study of the citation distribution. *The European Physical Journal B - Condensed Matter and Complex Systems 4*, 2 (1998), 131–134.

[180] Reichardt, J., and Bornholdt, S. Detecting fuzzy community structures in complex networks with a potts model. *Phys. Rev. Lett. 93*, 218701 (2004).

[181] Reichardt, J., and Bornholdt, S. Statistical mechanics of community detection. *Physical Review E 74*, 1 (2006), 016110.

[182] Ronhovde, P., and Nussinov, Z. Multiresolution community detection for megascale networks by information-based replica correlations. *Phys. Rev. E 80* (Jul 2009), 016109.

[183] Ronhovde, P., and Nussinov, Z. Local resolution-limit-free potts model for community detection. *Phys. Rev. E 81* (Apr 2010), 046114.

[184] Rubinov, M., and Sporns, O. Complex network measures of brain connectivity: Uses and interpretations. *NeuroImage 52*, 3 (2010), 1059 – 1069. Computational Models of the Brain.

[185] Santos, F. C., Santos, M. D., and Pacheco, J. M. Social diversity promotes the emergence of cooperation in public goods games. *Nature 454*, 7201 (07 2008), 213–216.

[186] Sanz, J., Xia, C.-Y., Meloni, S., and Moreno, Y. Dynamics of interacting diseases. *Phys. Rev. X 4* (Oct 2014), 041005.

[187] Saumell-Mendiola, A., Serrano, M. A., and Boguñá, M. Epidemic spreading on interconnected networks. *Phys. Rev. E 86* (Aug 2012), 026106.

[188] Schuetz, P., and Caflisch, A. Efficient modularity optimization by multistep greedy algorithm and vertex mover refinement. *Physical Review E 77*, 4 (2008), 046112.

[189] Scott, J. *Social network analysis*. Sage, 2012.

[190] Seeley, W. W., Crawford, R. K., Zhou, J., Miller, B. L., and Greicius, M. D. Neurodegenerative diseases target large-scale human brain networks. *Neuron 62*, 1 (2009), 42 – 52.

[191] Segal, M., and Manor, D. Confocal microscopic imaging of [Ca2+]i in cultured rat hippocampal neurons following exposure to n-methyl-d-aspartate. *The Journal of Physiology 448*, 1 (1992), 655–676.

[192] Serrano, M. A. Rich-club vs rich-multipolarization phenomena in weighted networks. *Phys. Rev. E 78* (Aug 2008), 026101.

[193] Singh, A., and Lesica, N. A. Incremental mutual information: A new method for characterizing the strength and dynamics of connections in neuronal circuits. *PLOS Comput Biol 6*, 12 (12 2010), e1001035.

[194] Smith, F. A., Lyons, S. K., Ernest, S. M., Jones, K. E., Kaufman, D. M., Dayan, T., Marquet, P. A., Brown, J. H., and Haskell, J. P. Body mass of late quaternary mammals: Ecological archives e084-094. *Ecology 84*, 12 (2003), 3403–3403.

[195] Solá, L., Romance, M., Criado, R., Flores, J., García del Amo, A., and Boccaletti, S. Eigenvector centrality of nodes in multiplex networks. *Chaos 23*, 3 (2013), –.

[196] Solé-Ribalta, A., De Domenico, M., Kouvaris, N. E., Díaz-Guilera, A., Gómez, S., and Arenas, A. Spectral properties of the laplacian of multiplex networks. *Phys. Rev. E 88* (Sep 2013), 032807.

[197] Son, S.-W., Bizhani, G., Christensen, C., Grassberger, P., and Paczuski, M. Percolation theory on interdependent networks based on epidemic spreading. *EPL (Europhysics Letters) 97*, 1 (2012), 16006.

[198] Soriano, J., Rodríguez Martínez, M., Tlusty, T., and Moses, E. Development of input connections in neural cultures. *Proceedings of the National Academy of Sciences 105*, 37 (2008), 13758–13763.

[199] Sporns, O. The human connectome: a complex network. *Annals of the New York Academy of Sciences 1224*, 1 (2011), 109–125.

[200] Sporns, O., Chialvo, D. R., Kaiser, M., and Hilgetag, C. C. Organization, development and function of complex brain networks. *Trends in Cognitive Sciences 8*, 9 (2004), 418 – 425.

[201] Sporns, O., and Zwi, J. The small world of the cerebral cortex. *Neu-*

*roinformatics 2*, 2 (2004), 145–162.

[202] Starnini, M., Baronchelli, A., Barrat, A., and Pastor-Satorras, R. Random walks on temporal networks. *Phys. Rev. E 85* (May 2012), 056115.

[203] Stetter, O., Battaglia, D., Soriano, J., and Geisel, T. Model-Free Reconstruction of Excitatory Neuronal Connectivity from Calcium Imaging Signals. *PLOS Comput Biol 8*, 8 (2012), e1002653.

[204] Strehl, A., and Ghosh, J. Cluster ensembles — a knowledge reuse framework for combining multiple partitions. *J. Mach. Learn. Res. 3* (Mar. 2002), 583–617.

[205] Sun, Y., Huang, Z., Yang, K., Liu, W., Xie, Y., Yuan, B., Zhang, W., and Jiang, X. Self-organizing circuit assembly through spatiotemporally coordinated neuronal migration within geometric constraints. *PLOS ONE 6*, 11 (11 2011), e28156.

[206] Szell, M., Lambiotte, R., and Thurner, S. Multirelational organization of large-scale social networks in an online world. *Proceedings of the National Academy of Sciences 107*, 31 (2010), 13636–13641.

[207] Takahashi, N., Takahara, Y., Ishikawa, D., Matsuki, N., and Ikegaya, Y. Functional multineuron calcium imaging for systems pharmacology. *Analytical and Bioanalytical Chemistry 398*, 1 (2010), 211–218.

[208] Tsai, C.-Y., Chang, M.-C., and I, L. Robustness and Variability of Pathways in the Spontaneous Synchronous Bursting of Clusterized Cortical Neuronal Networks In vitro. *Journal of the Physical Society of Japan 77*, 8 (Aug. 2008), 084803.

[209] Valdano, E., Ferreri, L., Poletto, C., and Colizza, V. Analytical computation of the epidemic threshold on temporal networks. *Phys. Rev. X 5* (Apr 2015), 021005.

[210] Valdano, E., Poletto, C., Giovannini, A., Palma, D., Savini, L., and Colizza, V. Predicting epidemic risk from past temporal contact data. *PLOS Comput Biol 11*, 3 (03 2015), e1004152.

[211] Verbrugge, L. M. Multiplexity in adult friendships. *Social Forces 57*, 4 (1979), 1286–1309.

[212] Vijayaraghavan, V. S., Noël, P.-A., Waagen, A., and D'Souza,

R. M. Growth dominates choice in network percolation. *Phys. Rev. E 88* (Sep 2013), 032141.

[213] Waltman, P. *Deterministic threshold models in the theory of epidemics*, vol. 1. Springer Science & Business Media, 2013.

[214] Wang, H., Li, Q., D'Agostino, G., Havlin, S., Stanley, H. E., and Van Mieghem, P. Effect of the interconnected network structure on the epidemic threshold. *Phys. Rev. E 88* (Aug 2013), 022801.

[215] Wasserman, S., and Faust, K. *Social Network Analysis: Methods and Applications*, 1 ed. No. 8 in Structural analysis in the social sciences. Cambridge University Press, 1994.

[216] Wasserman, S., and Faust, K. *Social network analysis: Methods and applications*, vol. 8. Cambridge university press, 1994.

[217] Watts, D. J. *Six Degrees: The Science of a Connected Age*, 1st ed. W. W. Norton & Company, 2003.

[218] Watts, D. J., and Strogatz, S. H. Collective dynamics of small-world networks. *Nature 393*, 6684 (06 1998), 440–442.

[219] Yodzis, P. Local trophodynamics and the interaction of marine mammals and fisheries in the benguela ecosystem. *Journal of Animal Ecology 67*, 4 (1998), 635–658.

[220] Yvon, C., Rubli, R., and Streit, J. Patterns of spontaneous activity in unstructured and minimally structured spinal networks in culture. *Experimental Brain Research 165*, 2 (2005), 139–151.

[221] Zachary, W. An information flow model for conflict and fission in small groups. *J. Anthropol. Res. 33* (1977), 452–473.

[222] Zhou, J., Gennatas, E. D., Kramer, J. H., Miller, B. L., and Seeley, W. W. Predicting regional neurodegeneration from the healthy brain functional connectome. *Neuron 73*, 6 (2012), 1216 – 1227.

[223] Zhou, S., and Mondragon, R. The rich-club phenomenon in the internet topology. *Communications Letters, IEEE 8*, 3 (March 2004), 180–182.