Optimisation and information-theoretic principles in multiplex networks

Andrea Santoro

A thesis submitted to the University of London for the degree of
Doctor of Philosophy

School of Mathematical Sciences
Queen Mary University of London
United Kingdom

September 2020

Declaration

I, Andrea Santoro, confirm that the research included within this thesis is my own work or that where it has been carried out in collaboration with, or supported by others, that this is duly acknowledged below and my contribution indicated. Previously published material is also acknowledged below.

I attest that I have exercised reasonable care to ensure that the work is original, and does not to the best of my knowledge break any UK law, infringe any third party's copyright or other Intellectual Property Right, or contain any confidential material.

I accept that the College has the right to use plagiarism detection software to check the electronic version of the thesis.

I confirm that this thesis has not been previously submitted for the award of a degree by this or any other university.

Signature: Date:

List of Publications

Many of the ideas and results presented in the sections and chapters of this thesis are taken from the following published works:

- A. Santoro, G. Nicosia, V. Latora, and V. Nicosia,
  Pareto optimality in multilayer network growth
  Phys. Rev. Lett. 121, 128302 (2018)[1].

- A. Santoro and V. Nicosia,
  Algorithmic complexity of multiplex networks
  Phys. Rev. X 10, 021069 (2020)[2].

- A. Santoro and V. Nicosia,
  Optimal percolation in correlated multilayer networks with overlap
  Phys. Rev. Research 2, 033122 (2020)[3].

Some of the work carried out during the PhD that is not included in this thesis is included in the following publications:

- A. Patanè, A. Santoro, P. Conca, G. Carapezza, A. La Magna, V. Romano, and G. Nicosia
  Multi-objective optimization and analysis for the design space exploration of analog circuits and solar cells
  Engineering Applications of Artificial Intelligence 62, 373 (2017) [4].

- A. Patanè*, A. Santoro*, V. Romano, A. La Magna, and G. Nicosia
  Enhancing quantum efficiency of thin-film silicon solar cells by Pareto optimality
  Journal of Global Optimization 72, 491(2018) [5].

- J. C. W. Billings, M. Hu, G. Lerda, A. N. Medvedev, F. Mottes, A. Onicas, A. Santoro, and G. Petri

Simplex2Vec embeddings for community detection in simplicial complexes

arXiv: 1906.09068 (2019) [6].

- A.Bassolas*, A. Santoro*, S. Sousa, S. Rognone, V. Nicosia

  Optimising the mitigation of epidemic spreading through targeted adoption of contact tracing apps

  In preparation

- A. Santoro, and V. Nicosia

  Maximal overlap of multiplex networks

  In preparation

Acknowledgments

As in all journeys, it comes the moment to nostalgically look back and draw some conclusions. I wish to thanks all the special people that have directly or indirectly supported me in this fantastic adventure.

First and foremost, my deepest thanks go to Enzo, who has been much more than a supervisor in all these years. He showed me the terrific and impervious path of the researcher since the very beginning, teaching me so much and treating me like a young colleague, rather than a student. I am also truly indebted to him for the countless opportunities of scientific and personal growth, allowing me to take risks, make errors, and grow into a better young researcher (or at least I hope so), and yet promptly ready to cheer me up during the moments of struggle and frustration. I remember a quote that always put a smile on my face: *If getting a new result was simple, we would have called it "recovering" rather than "research".* Special thanks go to Vito, for making me feel part of the group and of the big QMUL family since my first visiting in 2014, but also for the numerous suggestions, Christmas parties, and discussions in Roccalumera with granitas at Mimmo. Since the beginning, I have felt included and supported. Equally, huge thanks go to Prof. Giuseppe Nicosia and Prof. Vittorio Romano, who gave me the incredible chance to work on several research projects since my first years of university, introducing me to the research environment and, most importantly, provided me with the suggestions and guidance I needed over several years. I will be forever indebted to them for all those years, for the fruitful collaborations, and for putting me in contact with the group of QMUL. I am also very grateful to Dr Lucas Lacasa, Dr Giovanni Petri, Dr Massimo Stella, and Dr Federico Battiston (in random order), for the many discussions, suggestions, feedbacks over the years that helped me a lot during my PhD in several circumstances. I am also grateful to Prof. Martin Rosvall and Dr. Richard Clegg to act as examiners for this thesis.

Along my journey, I had the opportunity to meet numerous friends who reshaped, in several ways, my concepts of home and family. I wish to thank Iac and Ali for their amazing friendship and support for all these London years (just remember that everything blossomed from a Pizza and a Torta della Nonna, edddaaaai!!!). Thanks again for every moment, dinner, trip (con "li vestiti fresci"), and squash matches we had together. I wish to thank Jacopo, Elif, Valeria,

Lilli, Paolo, Moreno, Valerio for the precious and hilarious moments together, for the countless beers and dinners. I have to thank Arianna, Andrea, Luca, and especially Nicola, with whom I shared numerous cooking sessions (along with the overrated Marilú cake), for the numerous London dinners and beers. I am also grateful to my friends in the maths department at Queen Mary University of London, who helped me in different ways while working in the office and with whom I shared many teas, coffees, beers over the years. A special thanks go to Mayank, Vangelis, Francesco, Owen, Sandro, Aleix, and Silvia. During my PhD, I had also the opportunity to attend several conferences, warm-ups and, most important, the Complex system summer school in Santa Fe. I wish to thank the friends I met there, in particular Konstantinos, Eleonora, Cedric, Zohar, Guillaume, Sanna. I am also thankful to all the friends from the SSC, with whom I have directly or indirectly crossed the path in London and that have changed me over the years, in particular, Ester, Claudia, Giampy, Valerio, Leonardo, Sara, Valentina, Claudio, Gabriele, and the group from the university, Saverio, Andrea Pat, Peppe, Matilde, Angela, Carmen. A deep thanks go to my SSC room mates, Alessandro, Andrea, Angelone, for the countless adventures, experiences, stories, and discussions over the last years.

Super huge thanks go to my flatmates Andreas, Jimena, and Fabio with whom I shared so many jokes, discussions, experiences, beers, dinners that cannot be described in short. More than flatmates, they have represented a marvellous family for me in all these four years in London, tolerating my ups and downs with patience, unconditional support and joy, which allowed me to "blossom in a vibrant environment".

Finally, a lovely thanks goes to Luisa who shared with me, in the most splendid way, this journey. Without you, I would be lost. This thesis is dedicated to my family, my uncles, and my grandparents. They have been a fundamental part of this fantastic and challenging period in London. Who I am today I owe in large part to them.

Abstract

The multiplex network paradigm has proven very helpful in the study of many real-world complex systems, by allowing to retain full information about all the different possible kinds of relationships among the elements of a system. As a result, new non-trivial structural patterns have been found in diverse multi-dimensional networked systems, from transportation networks to the human brain. However, the analysis of multiplex structural and dynamical properties often requires more sophisticated algorithms and takes longer time to run compared to traditional single network methods. As a consequence, relying on a multiplex formulation should be the outcome of a trade-off between the level of information and the resources required to store it.

In the first part of the thesis, we address the problem of quantifying and comparing the amount of information contained in multiplex networks. We propose an algorithmic information-theoretic approach to evaluate the complexity of multiplex networks, by assessing to which extent a given multiplex representation of a system is more informative than a single-layer graph. Then, we demonstrate that the same measure is able to detect redundancy in a multiplex network and to obtain meaningful lower-dimensional representations of a system. We finally show that such method allows us to retain most of the structural complexity of the original system as well as the salient characteristics determining the behaviour of dynamical processes happening on it.

In the second part of the thesis, we shift the focus to the modelling and analysis of some structural features of real-world multiplex systems throughout optimisation principles. We demonstrate that Pareto optimal principles provide remarkable tools not only to model real-world multiplex transportation systems but also to characterise the robustness of multiplex systems against targeted attacks in the context of optimal percolation.

Contents

List of Figures

List of Tables

Introduction

Several natural systems constantly amaze us for their impressive complexity. Even the simplest forms of life, such as bacteria or yeast, rely upon hundreds of biochemical reactions that allow them to perform their essential functions. Millions of cells interact with each other in larger organisms to achieve more complex tasks. A typical example is the human brain [8], made up of approximately 10^{15} connections [9] among 100 billions of neurons [10]. However, complexity does not pertain to biological systems only. Power grids, transportation systems, or urban areas are some examples of man-made systems that exhibit a non-trivial complex structure. These systems are all paradigmatic examples of what is nowadays known as a complex system [11, 12], i.e. a system whose collective behaviour cannot be derived from the mere knowledge of the system's components. The advent of Network Science has completely shaken our way of analysing and modelling real-world complex systems, by describing them in terms of networks, where nodes represent the individual units (or components), and two nodes are connected by a link if the two units interact with each other [11, 13].

The first analyses of real-world systems mostly mapped all these systems to binary networks (i.e. networks in which links are either present or absent), therefore providing a robust mathematical framework to study complex phenomena. Over the years, network scientists realised that such a description was not enough. They gradually included weights [14, 15], and time [16–18] into the network representation, which helped to characterise new and known phenomena [14, 19, 20],

such as epidemics [21], epileptic seizures in neuroscience [22, 23], social contagion [24], and to discover disease biomarkers in medicine and biology [25–27]. The evolution of the spreading models over the last few years gives us a clear example of the tremendous impact of network science on the analysis and prediction of the diffusion of epidemics around the world [21, 28, 29]. For example, this is the case of the Zika virus [30] and the current COVID-19 pandemic [31–35]. In this case, advanced numerical models are used to forecast how the virus would spread and therefore to organise possible countermeasures to prevent extensive and even more dangerous outbreaks.

Transportation, social, or ecological systems cannot however be characterised solely by weighted connections among the basic units of a network. These systems are intrinsically intertwined by several interactions that take place among their components. A clear example is given by social systems, where people can interact with their acquaintances throughout several communication channels including face-to-face interactions, phone calls, emails, Facebook, Instagram or Twitter. In transportation networks, we can distinguish between different means of transportation including bus, coaches, metro, and train. To properly model these systems, *multilayer networks*, or *multiplex networks*, have rapidly become popular over the last few years [36–38]. These are relational systems whose units are connected by different relationships, with links of distinct types occurring at comparable time scales embedded in different layers. In particular, by allowing to retain full information about all possible kinds of relationships among the elements of a system, the multiplex framework represents an unprecedented opportunity to unveil new emerging phenomena [39–46]. For instance, new investigation and analysis in economics [47–50], biology [51, 52], and neuroscience [53–55], as well as in transportation [56, 57] and social sciences [58, 59], have benefited from a multilayered representation.

Nevertheless, all that glitters is not gold. Most of the analyses and models relying on the multiplex representation require more memory and advanced algorithms when compared to the single-layer network representation. More importantly, not all the available levels of interaction

among the constituents of a complex system have the same importance and some of them might be irrelevant, redundant, or uninformative, with respect to the overall structure of the system [60]. Therefore, it is essential to find the trade-off between the insights attainable by preserving full information about a system and the resources required to process and store these data. Some recent studies have also shown that the multilayer version of some dynamical processes cannot be reduced to the corresponding single-layer process on any simple combination of the existing layers [41]. Nevertheless, determining whether a lower-dimensional multilayer network can exhibit the same structural and dynamical richness of the full multilayer graph is still an open question. The recent attempt to formalise multilayer dimensionality reduction in terms of a quantum information problem [60] represents a concrete step in that direction. However, such a method still suffers from numerous shortcomings: (i) it is not designed to reduce a multiplex network with several layers into just a single-layer network; (ii) there is no evidence that the structural and dynamical properties of the lower-dimensional representations identified by the method are similar to the properties of the original multilayer system; (iii) it is relatively difficult to pinpoint a specific graph property as responsible for changes in the results of the method; (iv) it is not possible to compare the amount of information encoded in two different multilayer systems having different number of nodes, edges, and layers. As a consequence, we still lack of a convincing method that solves all the aforementioned issues and, more importantly, quantifies the amount of information contained in a multilayer network model. As we shall detail in Chapter 3, we propose a new methodology which tackles most of the points mentioned above and represents one of the main contributions of this thesis.

At the same time, the mere analysis of data (and its relative importance) does not always provide a comprehensive overview of the phenomena under analysis. In this regard, models are crucial to understand the reality of systems and the processes that govern them [61]. Empirical data and real observations add constraints on the model and lead to identify the most important parameters and processes. Both these ingredients, data and models, are therefore essential guides

to speed up the process for a complete characterisation of a phenomenon. Some of the first mathematical models proposed in network science [62–64] aimed at reproducing certain structural properties observed in real-world systems, such as the presence of clustering, the small average distance between nodes, and the degree distribution [11, 65]. In particular, a class of these models relies on the *rich gets richer* principle [64, 66], which allows to obtain power-law degree distributions using the so-called linear preferential attachment mechanism in which newly arriving nodes select and link existing nodes with a probability linearly proportional to their degrees. By contrast, other models are based on the assumptions that the interactions among the basic units of many real-world systems are often subject to different types of concurrent, and sometimes competing, constraints and objectives, such as the availability of energy and resources, or the overall efficiency of the resulting structure. It is therefore reasonable to assume that the systems that we observe today are the result of a delicate balance between contrasting forces, which can be modelled by means of an underlying optimisation process under a set of constraints [67–71]. Even the emergence of scale-free networks can be explained by simple optimization mechanisms [72–76], while it has been found that many of the properties of biological networks result from the simultaneous optimization of several concurrent cost functions [22, 77–83]. However, multi-objective optimisation theory has not yet been linked to any multilayer network representation of real-world systems. As a result, in Chapter 5.4.3, we fill this gap and provide the first multilayer network model, entirely based on multi-objective optimisation principles, which reproduces some of the structural measures observed in real-world transportation systems. We then proceed, in Chapter 6.3, to show how multi-objective optimisation tools provide an ideal framework in optimal percolation theory for multiplex networks and allow us to generalise existing single-layer targeted attack strategies in the context of multiplex systems.

Main contributions

In this thesis, we propose an algorithmic information-theoretic method to assess if and when a multilayer network representation provides a qualitatively better model than the classical single-layer aggregated network. We show through extensive numerical simulations that our new method accounts for most of the limitations found in existing methods, therefore providing a new state-of-the-art approach to quantify and compress the information within a multilayer network representation. In addition, we propose a new framework, based on optimality principles, which links multi-objective optimisation theory with multilayer networks. We first model the formation of multilayer transportation networks as a multi-objective optimisation process, and show that our model reproduces some of the structural patterns observed in real-world systems as diverse as airline, train, and bus networks. Finally, we provide evidence that the multi-objective optimisation tools can be further used in the context of optimal percolation in multiplex networks. In short, the contribution of this thesis to the field of complex systems is twofold.

On the one hand, we propose a definition of complexity for multiplex networks, rooted in algorithmic information theory, to quantitatively determine when the information content of a multilayer network is more informative than any of its layers or compressed single-layer representations. We then demonstrate, throughout numerical simulations, that our complexity measure can be used to *(i)* compare the complexity of different multiplex systems, *(ii)* obtain low-dimensional representations of multidimensional systems, *(iii)* cluster multilayer networks into a small set of meaningful superfamilies, and *(iv)* detect tipping points in the evolution of different time-varying multilayer graphs. In particular, in the context of obtaining low-dimensional representations of multidimensional systems, we show that our methodology outperforms the current state-of-the-art methods and provides, among other things, the first empirical evidence that the lower-dimensional representations, identified by our method, retain most of the structural and dynamical properties of the original multilayer system.

On the other hand, we revise the concept of Pareto optimality, which naturally emerges in multi-objective optimisation theory and evolution, in the context of multilayer networks. We introduce the first growth model, which links multi-objective optimisation tools with multilayer network theory, and show that such combination allows us to obtain simple but effective explanations for the evolution of real-world transportation networks. We further demonstrate how Pareto-optimal principles provide an extremely advantageous framework to characterise the robustness of many real-world multiplex systems against targeted attacks. In this regard, we propose new generalisations of existing single-layer targeted attack strategies for multiplex systems and show that these generalisations outperform all the state-of-the-art methods when applied to synthetic duplex networks.

Outline

This thesis is composed of six chapters, divided in two parts.

- **Part one: Information theory and networks**
  - *Chapter one* provides an introduction and the background concepts of information theory and complexity. In particular, we give the formal definition of Shannon entropy, Jensen-Shannon divergence, and we introduce some important concepts related to algorithmic complexity;
  - *Chapter two* gives a general overview of the recent framework of multilayer networks presenting the basic definitions and the main structural measures that we will consider throughout the chapters. We then revise some of the information-theoretic approaches recently proposed for single and multilayer networks, with a focus on the so-called "reducibility problem";
  - *Chapter three* presents our main contribution to the field, by proposing a new mea-

sure of complexity for multiplex networks rooted in algorithmic information theory. We demonstrate that such a definition can not only be employed to characterise the complexity of real-world multiplex systems but also to obtain lower-dimensional representations of a multiplex network. This chapter is entirely based on our work of Ref. [2];

- **Part two: Optimisation in multiplex networks**
  - *Chapter four* contains a general overview of multi-objective optimisation problems and lays the basis for the concept of Pareto optimality, on which we will extensively rely in the subsequent chapters. We also revise some classical and evolutionary methods that are commonly employed to solve this class of problems, highlighting a few metrics of performance used to assess and compare the solutions found by distinct methods;
  - *Chapter five* discusses different classes of multiplex models introduced in the last years, with a focus on null models, randomisation algorithms, and growth models. In the last section, we propose a new multiplex model based on Pareto optimality, which is able to reproduce quite accurately the structural properties of several transportation systems and provide novel efficient strategies for the formation of new routes. This section draws upon our work of Ref. [1];
  - *Chapter six* provides a general overview of several recent findings on random percolation processes on multiplex networks. We then move on to optimal multiplex percolation, showing that Pareto-optimal principles provide an ideal framework to generalise single-layer optimal percolation strategies to the case of multiplex networks. As a proof of concept, we study the optimal percolation problem in correlated multiplex with overlap. This latter section is mostly based on our work of Ref. [3].

Part I

Information theory and networks

Chapter 1

Information theory & complexity

Since Shannon's seminal paper in 1948 [84], information theory has proven to be a revolutionary field with an enormous impact on a variety of different areas, including communication theory, mathematics, physics, cryptography, and engineering. The impact of information theory has been crucial not only for the invention of compact discs and lossless data compressions, but also for the development of mobile phones and the Internet. Nowadays, with the increasing availability of new detailed data sets, information-theoretic approaches provide unique tools to pre-process and investigate many real-world phenomena. In the field of complex networks, for instance, information-theoretic tools have been extremely beneficial in developing new synthetic descriptors to characterise the components of large systems, such as social, technological, and biological networks. At the same time, however, information theory has laid the foundation for more theoretical areas, such as the emerging field of complexity.

In this chapter, we first introduce some basic definitions of information theory and then provide a brief overview of the concept of complexity. Finally, we review some of the notions of algorithmic information theory that will be used later in this thesis.

1.1 Basic definitions

1.1.1 Shannon Entropy

The concept of uncertainty and how to formally measure it was rigorously analysed for the first time in the field of classical probability theory by Claude Shannon [84]. The original definition of entropy was conceived within his theory of communication and it represents a fundamental mathematical limit in the source coding theorem [84]. To introduce the formal definition, let us consider a discrete random variable X defined in an alphabet or support $\mathcal{X}$ and a probability mass function $p(x) = \Pr\{X = x\}, x \in \mathcal{X}$.

Definition 1.1.1 (Shannon entropy)

The **Shannon entropy** *of a discrete random variable is defined as [85, 86]:*

$$H(X) = -\sum_{x\in\mathcal{X}} p(x)\,\log p(x) = E\left[\log \frac{1}{p(x)}\right] \tag{1.1}$$

where we denote expectation of a random variable by E.

If the base of the log function is 2, then the Shannon entropy is measured in *bits*. Alternatively, if the base of the logarithm is e, the Shannon entropy is instead measured in *nats*. Observe that when $p(x) = 0$, $H(X)$ is not defined. However, it is common to consider $0 \log 0 = 0$, which is justified by continuity (i.e employing l'Hospital's rule for indeterminate forms [87]). In particular, the Shannon entropy $H(X)$ of a random variable is a measure of the amount of information required to describe that random variable on average. As it is clear from Eq. (1.1), the Shannon entropy does not depend on the actual values of the discrete random variable X, but instead it only depends on the probability distribution $p(x)$ [85]. Remarkably, it is possible to axiomatically derive the functional form of the Shannon entropy by considering a small set of axioms that the entropy function of a random variable must satisfy (see [87–89] for several

axiomatic derivations and further discussions on this topic). Besides the direct interpretation in terms of expectation of a discrete random variable X, the Shannon entropy can also be regarded as the minimal expected number of binary questions ("yes" and "no") required to determine the value of the discrete random variable X.

The generalisation of the definition of entropy for two (or more) random variables comes naturally from Eq. (1.1). A number of useful definitions can be introduced when dealing with more than one random variable [85].

Definition 1.1.2 (Joint entropy)

Let us consider the pair of discrete random variables (X, Y), defined in the support set $\mathcal{X}$ and $\mathcal{Y}$ respectively, with joint probability distribution $p(x, y)$ [usually indicated by $(X, Y) \sim p(x, y)$]. We then define the **joint entropy** *as:*

$$H(X,Y) = -\sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} p(x,y)\log p(x,y) = -E\left[\log p(x,y)\right]. \tag{1.2}$$

It is also useful to define the conditional entropy [85], defined as:

Definition 1.1.3 (Conditional entropy)

If $(X, Y) \sim p(x, y)$, the **conditional entropy** *is defined as [85, 86]:*

$$H(Y|X) = -\sum_{x\in\mathcal{X},y\in\mathcal{Y}} p(x,y)\log\frac{p(x,y)}{p(x)}. \tag{1.3}$$

Alternatively, we can express the conditional entropy as [85]:

$$\begin{aligned} H(Y|X) &= \sum_{x\in\mathcal{X}} p(x)H(Y|X=x) \\ &= -\sum_{x\in\mathcal{X}} p(x)\sum_{y\in\mathcal{Y}} p(y|x)\log p(y|x) \\ &= -\sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} p(x,y)\log p(y|x) \\ &= -E\left[\log p(Y|X)\right] \end{aligned} \tag{1.4}$$

where $(X, Y) \sim p(x, y)$. In other words, the conditional entropy quantifies the amount of information requested to describe Y when the value of the random variable X is known. In general $H(X|Y) \neq H(Y|X)$ [85], but instead it is valid that $H(X) - H(X|Y) = H(Y) - H(Y|X)$. The latter relationship can be easily proven by considering the following identity:

$$\log p(X, Y) = \log p(Y|X) \ \log p(X) = \log p(X|Y) \ \log p(Y) \tag{1.5}$$

and by taking the expectation on both sides in order to obtain:

$$H(X, Y) = H(X) + H(Y|X) = H(Y) + H(X|Y). \tag{1.6}$$

From the definitions given above, it is also possible to derive some interesting properties that lay the foundations for more important results related to the independence bound on the Shannon entropy [85, 88, 90], such as:

- $H(X) \geq 0$
- $H(X, Y) \leq H(X) + H(Y)$
- $0 \leq H(X|Y) \leq H(X)$.

Let us now consider a collection of random variables $X_1, X_2, \ldots, X_n$ drawn according to the joint probability $p(x_1, x_2, \ldots, x_n)$. Then we have:

$$H(X_1, X_2, \ldots, X_n) = \sum_{i=1}^{n} H(X_i|X_{i-1}, \ldots, X_1) \tag{1.7}$$

which can be proven by considering the identity $p\left(x_1, \ldots, x_n\right) = \prod_{i=1}^{n} p\left(x_i|x_{i-1}, \ldots, x_1\right)$, and

evaluating the relationship [85]:

$$\begin{aligned} H(X_1,\ldots,X_n) &= -\sum_{x_1,x_2,\ldots,x_n} p\left(x_1,\ldots,x_n\right)\log p\left(x_1,\ldots,x_n\right) \\ &= -\sum_{x_1,x_2,\ldots,x_n} p\left(x_1,\ldots,x_n\right)\log\left[\prod_{i=1}^{n} p\left(x_i|x_{i-1},\ldots,x_1\right)\right] \\ &= -\sum_{x_1,x_2,\ldots,x_n}\sum_{i=1}^{n} p\left(x_1,\ldots,x_n\right)\log\left[p\left(x_i|x_{i-1},\ldots,x_1\right)\right] \\ &= -\sum_{i=1}^{n}\sum_{x_1,x_2,\ldots,x_i} p\left(x_1,\ldots,x_i\right)\log\left[p\left(x_i|x_{i-1},\ldots,x_1\right)\right] \\ &= \sum_{i=1}^{n} H(X_i|X_{i-1},\ldots,X_1). \end{aligned} \tag{1.8}$$

With the previous relationships in mind, we can now provide a very important bound on the joint entropy [85], i.e.:

$$H(X_1,X_2,\ldots,X_n)\leq\sum_{i=1}^{n} H(X_i) \tag{1.9}$$

where the equal sign holds if and only if the X_i are statistically independent. In particular, we have that:

$$H(X_1,\ldots,X_n)=\sum_{i=1}^{n} H(X_i|X_{i-1},\ldots,X_1)\leq\sum_{i} H(X_i), \tag{1.10}$$

where the last inequality follows directly from the condition $H(X|Y)\leq H(X)$.

1.1.2 Mutual Information

We additionally introduce the *mutual information* or *information gain*, which is an information-theoretic measure to quantify the reduction of uncertainty of a random variable X due to the knowledge of Y.

Definition 1.1.4 (Mutual information)

The **mutual information** *between two random variables is defined as [85, 91]:*

$$I(X;Y) = \sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} p(x,y)\log\frac{p(x,y)}{p(x)p(y)} \tag{1.11}$$

where $p(x,y)$ is the joint probability mass function.

The mutual information can be rewritten as a function of the entropies of the random variables as [85]:

$$\begin{aligned} I(X;Y) &= \sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} p(x,y)\log\frac{p(x|y)}{p(x)} \\ &= -\sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} p(x,y)\log p(x) + \sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} p(x,y)\log p(x|y) \\ &= -\sum_{x\in\mathcal{X}} p(x)\log p(x) - \left(-\sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} p(x,y)\log p(x|y)\right) \\ &= H(X) - H(X|Y). \end{aligned} \tag{1.12}$$

In addition, the mutual information possesses a number of different relationships, which are often related to the Shannon entropy. Here, we just mention some of these properties without providing proofs (for a detailed discussion see Ref. [85]):

1. $I(X;Y) = H(X) - H(X|Y) = H(Y) - H(Y|X)$
2. $I(X;Y) = H(X) + H(Y) - H(X,Y)$
3. $I(X;X) = H(X)$
4. $I(X;Y) \geq 0$
5. $I(X;Y) = I(Y;X)$.

As mentioned earlier, the mutual information quantifies the reduction of uncertainty of a random variable due to the knowledge of another one. For instance, the mutual information between two independent random variables X and Y is $I(X;Y) = 0$, since knowing one of the two variables

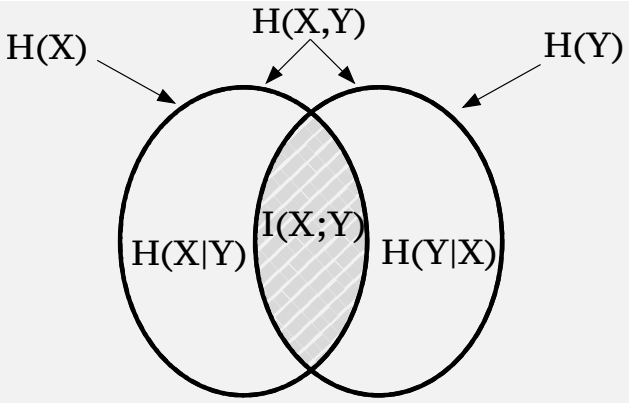

Figure 1.1. ***Graphic representation of mutual information.*** *Venn diagram representation of the relationships between entropy and mutual information.*

does not give any information about the other and vice versa. We report a pictorial representation of the mutual information using Venn diagrams in Figure 1.1. Notice that the representation of mutual information in terms of the Venn diagrams is straightforward only when considering two random variables. Indeed, the natural generalisation of the mutual information with more than two random variables can lead to certain overlapping areas which may correspond to negative values [91–95]. This is one of the reasons why there is still no consensus on a general extension of information theory for characterising the structure of multivariate interactions, despite numerous attempts [92, 93, 96, 97]. Nevertheless, applications of the mutual information in real-world problems span different disciplines. We only highlight new methods for comparing the similarity between clusterings [98, 99], and features selection in the field of machine learning [100, 101].

1.1.3 Kullback-Leibler divergence

Another important quantity that will be used later in this thesis is the concept of *relative entropy* or *Kullback-Leibler divergence* [85, 102, 103].

Definition 1.1.5 (Kullback-Leibler divergence)

The **Kullback-Leibler divergence** *(KL) between two probability distributions* $p(x)$ *and* $q(x)$

is defined as [85, 91]:

$$KL(p||q) = \sum_{x \in \mathcal{X}} p(x) \log \frac{p(x)}{q(x)} \tag{1.13}$$

Depending on the context, the measure is also known under several other names, such as *information divergence*, *cross entropy* or *Kullback-Leibler distance*. This represents a measure of inefficiency when assuming that the distribution is q when the true distribution is p [85]. Here it is also common to assume the following conventions justified by continuity: (i) $0 \log \frac{0}{0} = 0$, (ii) $0 \log \frac{0}{q} = 0$, and (iii) $p \log \frac{p}{0} = \infty$. Two of the main properties of the KL divergence are:

1. $KL(p||q) \geq 0 \quad \forall p, q$, while $KL(p||q) = 0$ iff $p(x) = q(x)$ (the proof of this property relies on Jensen's inequality [85, 104]);
2. $KL(p||q) \neq KL(q||p)$.

Although the KL divergence is often used for measuring the distance between two distribution, it is not a proper metric, since it does not satisfy the triangle inequality and the symmetric property [105]. Despite its simple definition, the usage of KL divergence has been steadily increasing in the last two decades in many practical contexts, assuming different connotations in a wide range of applications. For instance, it has been used to track "surprise" in cognitive science [106], visual search tasks [107, 108], study of language [109, 110], and text analysis [111–113]. However, KL divergence still lacks some useful properties we might desire for a measure, namely, symmetry and finiteness properties. For this reason, in some interesting applications discussed later in this thesis, it will be useful to consider the *Jensen-Shannon divergence* [114]. This is a measure that corresponds to a symmetrised and finite version of the KL divergence.

Definition 1.1.6 (Jensen-Shannon divergence)

The **Jensen-Shannon divergence** *(JSD) between two probability distributions $p(x)$ and $q(x)$ is defined as [85, 91]:*

$$JSD(p||q) = \frac{1}{2} KL(p||m) + \frac{1}{2} KL(q||m) \tag{1.14}$$

where $m = \frac{1}{2}(p + q)$.

In general, for two probability distributions, the $JSD(p||q)$ is bounded between 0 and 1 when considering the base 2 logarithm, or between 0 and log(2) using the natural logarithm [114].

1.2 Complexity

With the advent of information theory, new interdisciplinary approaches aimed at investigating the structure and organisations of real-world systems started to take place at the crossroads of mathematics, physics, chemistry, biology, and computer science. The study of complex systems and their emergent "complex" behaviours represent a paradigmatic example. To date, we can intuitively describe the complexity of a system or dynamical process as the degree to which components engage in organized structured interactions. In particular, high values of complexity are usually associated with systems that show a mixture of order and disorder and, at the same time, are able to generate emergent phenomena [115].

But how can we formally define *complexity*? A satisfactory answer to this question, along with a unified theory, are still subject to debate. A nice quote from Melanie Mitchell [116] can clarify more on this: *'there is not yet a single science of complexity but rather several different sciences of complexity with different notions of what complexity means'*. In the last eighty years, several contributions coming from a wide range of disciplines proposed consistent and formal definitions of complexity. One of the main underlying fundamental questions entails differentiating ordered from disordered systems. Along this line, in 2001 [117], Lloyd posed three different questions that a measure of complexity of an object should address [116, 117]:

- How hard is it to describe?
- How hard is it to create?
- What is its degree of organization?

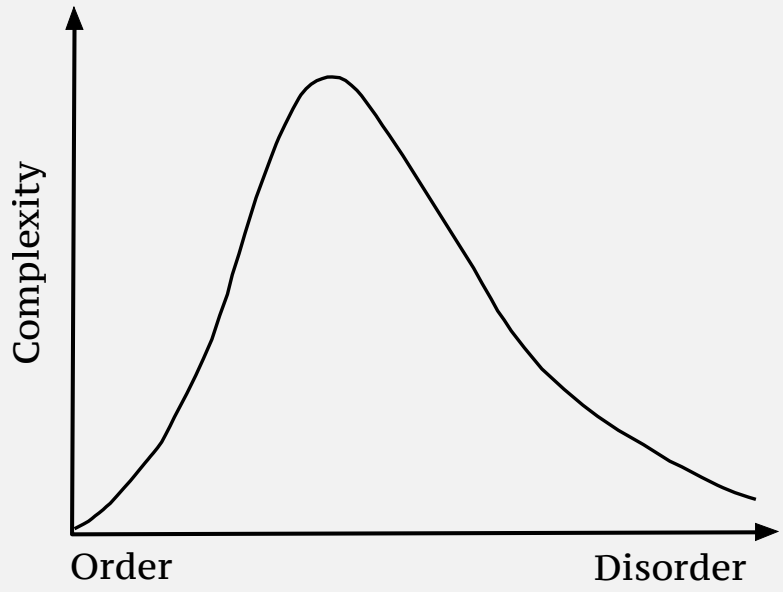

Figure 1.2. ***Behaviour of a "good" complexity measure.*** *Complexity as a function of order and disorder. Reprinted from [118].*

More than 30 definitions of complexity currently exist [117]. However, a generally accepted quantitative measure linking complexity with order and disorder is still missing (see Fig. 1.2 for a pictorial representation of the property that a good complexity measure should possess). Interestingly, the behaviour presented in Fig. 1.2 somehow reflects our intuitive concept of complexity [116]. *The most complex entities are not the most ordered or random ones but somewhere in between* [116]. This is often confirmed by looking at several real-world systems, such as the organisation of the axons in the human brain or the distribution of the nucleotides in our genome. They are neither entirely random nor completely ordered. In the following, we review some of the definitions of complexity mainly coming from the fields of physics, computation, and information theory, trying to follow a chronological order rather than dividing them in macro-families.

Warren Weaver was one of the first scientists who preliminary addressed the concept of complexity. In one of his works [119], he formalised the conceptual distinction between systems made of large number of elements with high degree of freedom (called *"disorganized complexity"*) and systems wherein the interactions between elements occur but are not random (named *"organised complexity"*). Phenomena belonging to the first class can be easily characterised by tools from

Figure 1.3. ***Example of complexity in natural ecosystems.*** *Colony of nomadic army ants in Costa Rican forest. In this case, the emerging phenomena cannot be characterised by the mere knowledge of its single elements (Reprinted with permission of Daniel Kronauer, 2019 Wildlife Photographer of the Year).*

probability and statistical mechanics, as for instance the temperature or the average velocity of particles in a perfect gas. By contrast, *'problems which involve dealing simultaneously with a sizeable number of factors which are interrelated into an organic whole are all problems of organized complexity'* [119]. As a consequence, the behaviour of the whole cannot be merely studied by investigating average quantities or properties of single elements (see for instance Fig. 1.3). Examples in this category include the evolution of living organisms, as well as the functioning of the DNA and genome. In the context of biological systems, for instance, how can we characterise the complexity of a certain genome when looking at the complete collection of nucleotides (A, C, G, and T for DNA genomes)?

One way is to rely on the Shannon entropy [84], introduced in Section 1.1.1. In this case, by putting the object to be analysed as a form of message (of some kind), the Shannon entropy of that object can be seen as the average information content a message source has for the receiver [116].

However, the entropy function assumes its maximum value when the message is completely random (uniform probability). Thus, a completely random genome containing an equal number of nucleotides would be extremely "complex" according to this definition. As a consequence, the Shannon entropy cannot be considered a genuinely "good" measure of complexity. In fact, as mentioned before, we intuitively expect a complex object to be neither completely random nor ordered but rather lie somewhere in between.

In the context of algorithmic computation and information theory, one of the most famous definitions of complexity was provided by Kolmogorov [120] and Chaitin [121]. This definition relies on a suitable model of computation, i.e. the Turing machine [122]. This represents a mathematical model of computation, which can be conceptually regarded as the simplest universal computer [123, 124]. The Turing machine is formed by three elements: (i) an infinitely long tape, divided into cells, where symbols can be written or read from, (ii) a moveable "head" which reads/writes symbols from/to the tape, and (iii) a set of instructions that tells the machine what to do next [85, 116]. In the late 1930s, Turing proved that it is possible to design a special Turing machine, called Universal Turing Machine ($\mathcal{U}$), which is able to emulate the behaviour of any other Turing machine [122].[1] With the Turing machine as a model of computation, the Kolmogorov-Chaitin complexity of a string of symbols is then defined as the shortest computer program that generates that string. This definition provides a way to quantify the amount of "randomness" contained in strings and paves the way for many other measures of complexity. To provide a concrete example, let us consider the following two strings made of 32 lowercase letters:

$$aeaeaeaeaeaeaeaeaeaeaeaeaeaeae$$

$$gdsahjoashgjawhrhtsldfhglshgdlld$$

[1] Inspired by Gödel's work [125], Alan Turing introduced the formal concept of Turing machine, called a-machine [122], to solve the "*Entscheidungsproblem*", which was (re)formulated by D. Hilbert in 1928 [126]. This is commonly known as *the halting problem* or simply to decide whether it exist a definite procedure that can decide whether or not a statement is true [122, 127, 128].

If we consider an English-like description, the first string can be simply represented as "*Print ae 16 times*", which has only 17 characters, while the second string does not have a simple description. Hence, the only way to represent it is by "*Print gdsahjoashgjawhrhtsldfhglshgdlld*", which has 38 characters. As a consequence, if we consider the shortest description length, the first string is less complex than the second one. However, the inconvenience of the Kolmogorov complexity is that of being a non-computable function [85, 129]. Formal proofs for this inconvenient feature can be found in Ref. [130]. We will provide an extensive discussion on algorithmic information theory and more details on the definition of Kolmogorov complexity in Section 1.3.

In the context of dynamical systems, one of the first measures used to quantify the complexity of an object is the *fractal dimension* [131, 132]. Briefly, *it represents the amount of copies of a self-similar object at each level of magnification of that object* [116]. Or, put another way, how much detail you see at all scales when looking trough all the copies of a self-similar object. A non-trivial example is given by the fractal dimension of a Koch snowflake [133]. Besides the difficulty in applying this definition to a wide range of practical cases, the amount of copies in a self-similar object is not the only thing that we would like to quantify when employing a complexity measure.

In the same spirit of the Kolmogorov complexity, when quantifying the amount of randomness in an ensemble of patterns, one interesting quantity to look at is the *effective measure complexity*. This metric was introduced by Grassberger [134] in 1986[2]. Given a sequence of symbols, it is defined as the minimum amount of information contained in a given part of the sequence that is needed to predict (i.e. in the sense of minimal uncertainty) the next symbol [134]. A similar, yet distinct computation theoretic approach was provided by Crutchfield and Young [136] with the concept of *statistical complexity*. In a nutshell, the statistical complexity measures the minimum amount of information about a past configuration (i.e. a stream of consecutive symbols) of a system that, on average, is requested to reproduce the configuration assumed by the system in

[2]Notice that the definition of effective complexity provided by Grassberg should not be confused with the more famous measure proposed by M. Gell-Mann and S. Lloyd in 1996 [135].

the future [116, 136]. Hence, to predict the behaviour of the future configuration, it is often necessary to construct a model that permits the emulation of the system's behaviour in a statistically indistinguishable way. Clearly, in the same spirit of the Occam's razor [85], a simpler computational class is always preferable over a higher level class. Within this definition, both completely ordered and completely disordered configurations would be identified as having low complexity. As an example, genomes made of either completely random or extremely periodical sequences of nucleotides hold a low value of statistical complexity. This is because the minimum amount of information required for describing these two configurations, namely, strings with the instruction "repeat pattern AT" or "random element from {A,T,C,G}", have a low information content.[3] However, in many contexts this quantity is not easy to compute, especially when considering systems that cannot be directly mapped to configurations made of stream of consecutive symbols.

Related to the concept of Kolmogorov-Chaitin complexity, Bennett [137] introduced the definition of *logical depth* as a measure of complexity for individual strings. The measure is rooted in information theory and computational complexity. It is defined as the minimal amount of information (in terms of memory and time) requested to construct the string. Unsurprisingly, the definition of logical depth shares many features with the Kolmogorov complexity (e.g. the non-computability) and is also based on a proper model of computation, i.e. the Turing machine. When applying logical depth to a general object, the main idea is to convert it into a binary string and then construct a "suitable" Turing machine that is able to generate that string. The logical depth is then defined as the total number of steps carried out by such a Turing machine to reconstruct the string. However, among all the possible "suitable" Turing machines, Bennett identifies the one having the lowest number of states as the one to be selected. Here, again, the main issue with this definition is that it cannot be applied to a wide range objects, or more in general, to natural phenomena. For the latter, in fact, there are no practical ways of finding

[3]Notice that within this definition, it is allowed to have random choices.

suitable Turing machines.

In the context of biological systems and evolution, Adami and Cerf [138] proposed another intriguing complexity measure for sequences, called *physical complexity*, which again relies on Kolmogorov complexity. Here, the main hypothesis is that the complexity of a string cannot be considered in isolation (e.g. using the Kolmogorov complexity), but must instead be linked to the information content included within its environment. This is because a specific sequence might encode a high amount of information about one environment (niche) while being completely random when compared to another [139]. Hence, the physical complexity of a string is defined as the Kolmogorov complexity that is shared between the string under consideration and some description of the environment in which that sequence needs to be interpreted. Clearly, this case also inherits the main drawbacks of the Kolmogorov complexity (i.e. the non-computability). As a consequence, Adami [138] proposed to use the mutual information between the ensemble of sequences and their respective environment to approximate the physical complexity. As we shall see in the next Chapter, we extensively rely on some of the complexity measures discussed so far to introduce a novel measure to quantify the complexity of multiplex networks.

1.3 Algorithmic information theory

In line with the observations provided in the previous section, another interesting approach comes at the interface of computer science and information theory. More precisely, in the late '60s Kolmogorov [120, 140], Solomonoff [141, 142], and Chaitin [143] started investigating the relationship between computation and information of computably generated objects, like strings, which has recently inspired new theoretical and algorithmic approaches for the analysis of many complex systems [144–146]. Indeed, one of the main problem related to the definition of entropy is that it relies on the probability distribution and it therefore depends on the level of "granularity"

chosen. For instance, if we considered the bit string of length n:

$$01010101010101010101010101010101$$

which has a regular pattern, there exist $\binom{n}{i}$ possible substrings that we can consider [144], with $i \in \{1, \ldots, n\}$. In particular, the Shannon entropy is maximal [equal to 1 bit] at the level of individual bits, i.e. same number of 0s and 1s. Conversely, if we consider the pattern of two-block bits, we obtain a minimal complexity since it only contains one (01) of the 4 possible symbols (11, 10, 01, 00). More precisely, the Shannon entropy of the string presented above, when considering patterns of two-block bits, is equal to zero while the maximum value attainable is 2 bits. To mitigate this problem, a possible solution consists on considering all possible substring lengths. This quantity corresponds to the so-called *block entropy* [84, 147].

With the definition of Turing machine as a model of computation, Kolmogorov went further. He formally defined the *Kolmogorov complexity* [also named *algorithmic complexity* or *algorithmic information content* (AIC)] of a string s as [85, 148]:

$$KC_{\mathcal{U}}(s) = \min\{|p| : \mathcal{U}(p) = s\} \tag{1.15}$$

where with $\mathcal{U}(p)$ we denote the output of the computer $\mathcal{U}$ when presented with a program p, while $|p|$ is the length of the computer program p. The algorithmic complexity therefore represents the minimum description length over all programs that print s and halt [85]. Notice that we can use $KC_{\mathcal{U}}(s)$ or $KC(s)$ interchangeably, since $KC_{\mathcal{U}}(s)$ depends on $\mathcal{U}$ up to a constant, thanks to the invariance theorem [85, 148]. Formally, the latter theorem states that $\exists c$ such that $|KC_{\mathcal{U}}(s) - KC_{\mathcal{A}}(s)| \leq c$, where c is a constant independent from the universal computer $\mathcal{U}$ and any other computer $\mathcal{A}$[4]. Notice that the constant c might result very big for short strings but, in general, we have convergence for the limit of long strings. Despite all these interesting

[4]Other equivalent models of computations, such as lambda calculus or μ-recursive functions, can in principle be used instead of the Turing machines.

properties, the Kolmogorov complexity has the technical inconvenience of being a non-computable function [85, 149]. This is a consequence of the negative answer for the *halting problem* [122, 126]. Indeed, the unique way to find the shortest computer program is to test all the possible programs and find the one with the shortest description. However, when testing all the programs, we have no effective way of telling whether or not all the programs will halt. Thus, we are not able to find the shortest program to print a given string s [85, 130]. In practice, an upper bound for the algorithmic complexity can be approximated by using compression algorithms. A common, yet debated approach [150] is to compress the string s using a certain compression algorithm, and to consider the sum of the length of the compressed string $\bar{s}$ and the length of the decompression routine as an estimate of $KC(s)$. In fact, it is possible to obtain s from the compressed string $\bar{s}$ by using the decompression routine d (also called "*inflate*") corresponding to the compression algorithm used to obtain $\bar{s}$. Hence, the concatenation of $\bar{s}$ and the decompression routine (i.e., $\bar{s}||d$) is a program able to generate s, and its length is an upper bound for $KC(s)$. However, notice that the latter approach should be mainly used to estimate the KC of long strings since it essentially captures statistical redundancies [150]. Indeed, there exists a relationship between algorithmic complexity and entropy. It can be shown that the expected value of the algorithmic complexity of a generic random sequence is "close" to the Shannon entropy (the formal proof relies on the Jensen and Kraft inequalities [85]). The Kolmogorov Complexity of a string s of length n is smaller or equal than $nH(s) + \mathcal{O}(n)$ [151, 152], where $H(s)$ is the Shannon entropy when considering the probability distribution obtained from frequencies of symbols in s. By contrast, it is necessary to rely on more refined approaches, such as the block decomposition method [153] or the coding theorem method [152], when accurately estimating the algorithmic complexity of short strings.

Closely related to Kolmogorov complexity is the concept of *algorithmic probability* or *Solomonoff probability* [85, 148]. First introduced in the '60s by Solomonoff [141, 142, 154], and further formalised by Levin [155], the idea is rooted in finding a mathematical formalisation for assigning

probability values to explain a given observation. Let us consider a random program p or, equivalently, a program obtained from a series of fair coin flips. If p is fed into a universal Turing machine, with a certain probability the program will output something meaningful. If so, will the output sequences look random? Interestingly, the probability distribution on the output strings is far from uniform. The *universal probability* of a string s is defined as [85, 148]:

$$\mathcal{P}_{\mathcal{U}}(s) = \Pr(\mathcal{U}(p) = s) = \sum_{p:\,\mathcal{U}(p)=s} 2^{-|p|} \tag{1.16}$$

where again $|p|$ is the length of the computer program p, which is randomly drawn as a sequence of fair coin flips. Such probability measure therefore induces a distribution over programs producing the string s, so that shorter programs are much more probable than longer ones. As it is clear from the definition, this is also in line with the idea behind Occam's Razor [85]. Again, there exist a deep connection between universal probability and Kolmogorov complexity, which is formally expressed by the *(algorithmic) coding theorem* [85, 148, 156]:

$$2^{-KC(s)} \leq P_{\mathcal{U}}(s) \leq c\, 2^{-KC(s)} \tag{1.17}$$

where c is a constant independent from s. Hence, the universal probability of a string s is determined essentially by its Kolmogorov complexity. Alternatively, we can also rewrite Eq. (1.17) as [144]:

$$|-\log_2 P_{\mathcal{U}}(s) - KC(s)| < \mathcal{O}(1). \tag{1.18}$$

Chapter 2

Information theory in single and multi-layer networks

In this chapter, we discuss how information-theoretic measures have been increasingly used in the field of network theory and complexity science, providing powerful tools to characterise real-world systems. We first provide a brief overview on the application of information theory in the context of single-layer networks. We then move on describing the salient aspects of the multilayer formulation, providing a general summary of some of the local and global properties introduced to characterise the units of multilayer systems. Finally, we review some of the recent information-theoretic techniques proposed in the context of multilayer system, focusing on the problem of finding lower-dimensional representations of a multiplex network that can exhibit the same structural and dynamical richness as the full multilayer graph.

2.1 Information-theoretic approaches in single-layer networks

The idea of measuring the information content of a graph was firstly explored in the '50s by Rashevsky, Trucco, and Mowshowitz [157–160]. Here, automorphism-based considerations relying on graph invariants, such as the number of nodes, links, or degree sequences, were examined to measure the information complexity of graphs. The graph invariants are properties that do not change under isomorphism. Within this context, for example, the automorphism-based metric developed in Ref. [158, 159] aims at evaluating the information content of a graph by looking at the Shannon entropy of its automorphism group (see the mathematical formulation in Ref. [159]). In other words, such measure tries to capture the symmetry structure of a graph. As a result, at one extreme there is the fully connected network, which has the full symmetric group and thus it holds zero information content. By contrast, graphs with only the identity carry the maximum information content according to this measure. With the advances in network science theory, a number of studies started investigating the information content of networks in terms of several topological features such as network regularity, the number of connected components, the shortest-path distance, and non-trivial symmetries in the graph [160–163]. On the one hand, these studies are at the foundation of ad-hoc entropy measures for characterising the relationship between disorder and complexity of networks [67, 164]. This implies being able to detect phase transitions [165] when the network topology changes. On the other hand, information theory techniques are extremely useful when we seek to quantify the similarities between networks in a meaningful manner [166–169]. This is because information-theoretic methods based on graph invariants are able to cope with networks having different size and number of links. Some of the most recent metrics are based on the Jensen-Shannon divergence [166–169] (introduced in Section 1.1.3) to quantify the dis/similarity among certain network-based distributions, as for instance, the shortest path length distributions, the spectrum of the adjacency or Laplacian matrix,

or on the communicability matrix [170]. Applications in this direction include the field of neuroscience [23], where the ability to compare the network structure of healthy and sick patients, or to distinguish the graph structure before and after a specific medical treatment is crucial. Other applications involve the process of classifying and distinguishing protein networks [171], or tracking the evolution of temporal networks [16].

Another popular approach comes from the field of statistical physics [172]. The original idea goes back to the 19th century with Maxwell, Boltzmann and Gibbs [173–175]. In this case, the statistical mechanics approach has been successfully employed to study gas theory as an alternative solution to the more difficult and complex kinetic theory models. In a similar fashion, statistical physics approaches are employed to investigate several real-world complex networks [172, 176]. Here, the first step entails identifying a set of graph observables, or static properties, usually extracted from the empirical observation of real-world systems. Examples of these observables are the number of links in the network, the number of triangles, the degree distribution, and so forth. These quantities have a role which is similar to energy in statistical mechanics. Subsequently, by using the maximum entropy principle [177, 178], it is possible to construct maximal unbiased ensembles of random graphs that hold certain sets of properties, e.g. all the random graphs in the ensemble share the same degree distribution. The application of this ensemble is twofold. On the one hand, when the microscopic configuration of an empirical network is known, the corresponding ensemble of random graphs provides a null model for assessing the significance of features present in the empirical network. On the other hand, when the microscopic configuration is unavailable, the statical ensemble provides the most probable configuration with unbiased assumptions [172].

The mathematical formalism relies on exponential random graphs models (ERGM), first introduced in Ref. [179, 180], and later presented in physical terms using a complex network framework by Park and Newman [181]. By starting from a set of empirical observations x_i of some real-world network $G^\star$, we can model the graph $G^\star$ by associating to it an ensemble of graphs $\mathcal{G}$ with

the same number of nodes, same type of links [172]. In addition, the ensemble $\mathcal{G}$ has to satisfy the constraints, which guarantee that all the graphs in the ensemble have the same specified properties of $G^\star$. Then, for each graph G in the ensemble $\mathcal{G}$, a probability $P(G)$ defines the occurrence of that graph within the ensemble. With the tools of information theory and statistical physics [177], this problem can be solved in statistical terms. Indeed, the probability distribution that provides the most unbiased estimation of the empirical system $G^\star$ is the one that maximises the Shannon entropy (or alternatively, in physical terms, the Gibbs entropy) [172, 181]:

$$S = -\sum_{G\in\mathcal{G}} P(G)\ln P(G) \tag{2.1}$$

subject to:

$$\sum_{G\in\mathcal{G}} P(G) = 1 \tag{2.2}$$

constrained to:

$$\begin{cases} x_i(G) = x_i \quad \forall\, G \in \mathcal{G} & \text{hard constraint} \quad (2.3a) \\ \displaystyle\sum_{G\in\mathcal{G}} P(G)x_i(G) = x_i & \text{soft constraint} \quad (2.3b) \end{cases}$$

where $x_i(G)$ represents the value of x_i in G. The first equation [Eq. 2.2] represents the normalisation condition, while the second one includes two possible kinds of constraint (i.e. macroscopic properties) that can be enforced on the system, namely, the hard and soft constraints. While the hard constraints are met exactly by each graph in the resulting ensemble, the soft constraints are instead realised as ensemble averages [i.e. every graph of the ensemble satisfies the soft constraints on average (weighted by probability)]. Interestingly, starting from ERGM and depending on the nature of the constraints imposed, we obtain different network models. For instance, by imposing hard constraints [Eq. (2.3a)], we obtain the so-called *microcanonical network ensemble*, which corresponds to the classical configuration model [182] when fixing the exact degree sequence as constraint. The microcanonical ensemble lays the foundation for null models that

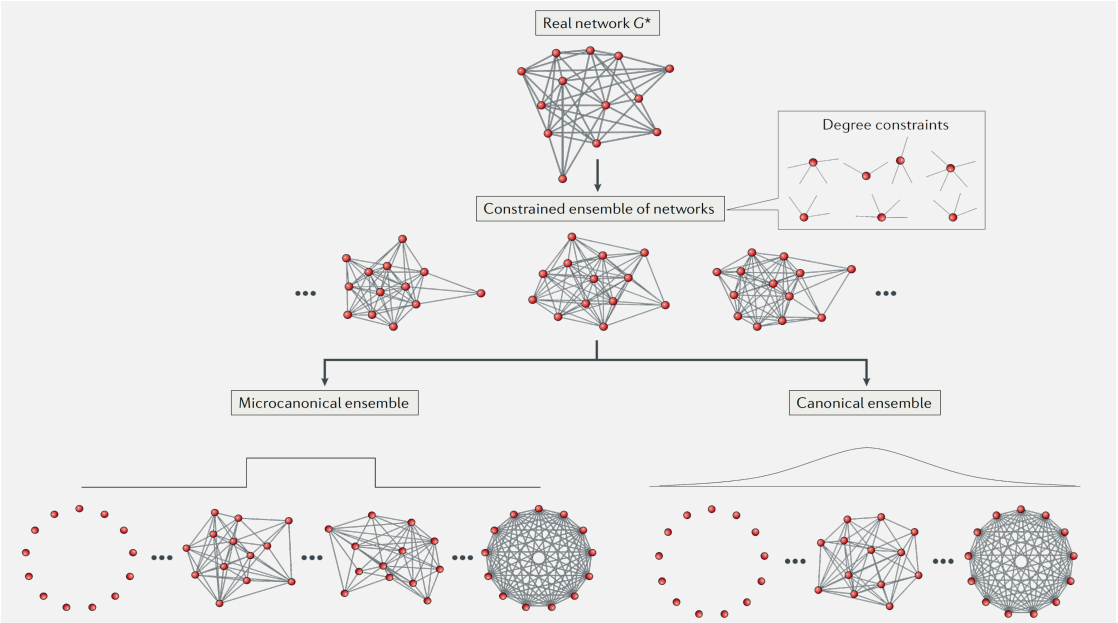

Figure 2.1. ***Statistical physics approach to obtain the microcanonical and canonical ensemble in complex networks.*** *The observations of the empirical network $G^{\star}$ provide the constraints x_i for the statistical ensemble obtained when maximising Eq. (2.1). In this representation, the degree sequence of the network $G^{\star}$ is imposed. The microcanonical ensemble is obtained as a results of imposing the exact degree sequence of $G^{\star}$ to each graph G of the statistical ensemble $\mathcal{G}$, i.e. Eq. (2.3a). As a consequence, the probability $P(G)$ is not null only when the constraints are exactly enforced. The canonical ensemble is instead obtained by constraining the expected degree value of each graph G within the ensemble $\mathcal{G}$ to be equal to the empirical observations x_i, i.e. Eq. (2.3b). Adapted from [172].*

are vital when assessing the significance of empirical patterns found in real-world networks. By contrast, if soft constraints are imposed, i.e. fixing the expected values of certain observables [Eq. (2.3b)], the so-called *canonical ensemble* is obtained. For instance, the Erdös-Rényi (ER) random model [62] can be derived throughout the ERGM formalism by constraining the expected number of total links [181]. In the thermodynamic limit the two ensembles, namely, the micro-canonical and canonical, turn out to be different [183–186], which does not match with the results of classical statistical physics. We shall consider this approach once more in Chapter 5.1.1 when introducing equilibrium models in multiplex networks.

Within the same framework, recent studies have also introduced several entropy measures (e.g. Shannon and Von Neumann entropy) associated with ensemble of networks to investigate the role of structural features in shaping a given real-world network [183, 184, 187, 188]. As an example,

the structural entropy introduced in Ref. [189] intuitively quantifies the entropy of an ensemble of uncorrelated undirected *simple networks* with given degree sequence, where with *simple networks* we refer to networks without self-links – links from a node i to i – or double links – two or more links between the same pair of nodes. This provides some insights into the role of degree distributions in networks. For instance, in Ref. [189], the author showed that, by using a statistical mechanics model, power-law degree distributions [11, 64, 190] are the more likely distributions associated with small values of structural entropy. By contrast, Poisson degree distributions are the most likely degree distribution of networks associated with high values of structural entropy.

At the beginning of the 21st century, another prolific line of research started focusing on quantifying the amount of information necessary to navigate in a network (e.g. between two streets in a city) [191, 192] and to send signals within a network [193]. To this end, the authors of Ref. [192] rely on random walks to examine the communication pathways within a complex network. More precisely, let us consider a graph G and denote with $p(r,s)$ the shortest path between two nodes r and s in the graph. Then, the probability of following this path for a random walk can be expressed as:

$$P[p(r,s)] = \frac{1}{k_r} \prod_{j \in p(r,s)} \frac{1}{k_j - 1} \tag{2.4}$$

where k_r is the degree of vertex r, while j counts all the nodes of the path $p(r,s)$ with the exclusion of r and s. Notice that the factor $k_j - 1$ on the denominator keeps track of the information gained when following the path, hence why the reduction of the number of links by one. Thus, to characterise the difficulty of navigation in networks, the authors of Ref. [192, 193] introduced the *search information* S defined as:

$$S(r,s) = -\log_2 \sum_{\{p(r,s)\}} P\left[p\left(r,s\right)\right] \tag{2.5}$$

where the sum runs over all the paths connecting nodes r and s. This entropy measure represents

the total information required to identify one of the shortest paths between r and s. Additionally, to characterise the overall complexity in navigating a network, the authors also introduced the average search information:

$$S = \frac{1}{N^2} \sum_{r,s} S(s,r) . \tag{2.6}$$

This metric has been used to show the difficulty in navigating city networks compared to their random counterparts [191]. The same authors have investigated other metrics to quantify the predictability of the message flow, and also to determine in which manner the organization of networks affects the navigability in complex topologies using analogous consideration [192, 194, 195]. More importantly, these works triggered new research directions aimed at quantifying the optimal navigability within complex networks with a limited amount of information [196–198]. Also in community detection methods [199], the synergetic combination between random walks and information-theoretic approaches represents the core of one of the popular community detection methods, namely, InfoMap [200–202].

When referring to a network with community structure, we shall refer to a network where the nodes of the system can be grouped into *modules* (also called *communities* or *clusters*) [11, 65, 199], such that nodes within the same cluster are more tightly connected than nodes belonging to two different clusters. In such cases we say that the networks have a *community* or *modular structure*. In particular, finding the communities can be useful not only to group together the nodes with similar functions in a network, but also to coarse-grain a large network so that is possible to draw it in a meaningful way (for comprehensive reviews on the topic see Refs. [199, 203, 204]). In the last twenty years of research, several community detection methods have been proposed [205–208]. Here, we just mention the two algorithms that represent the state of the art in the analysis of community structure, i.e. the Louvain [209] and Infomap algorithm [201]. While the former algorithm uses a simple local scheme to optimise modularity [206], and allows the existence of modules at different scales to be revealed, the latter is not based on the modularity but on an information-theoretic framework. More precisely, Infomap is rooted in information-

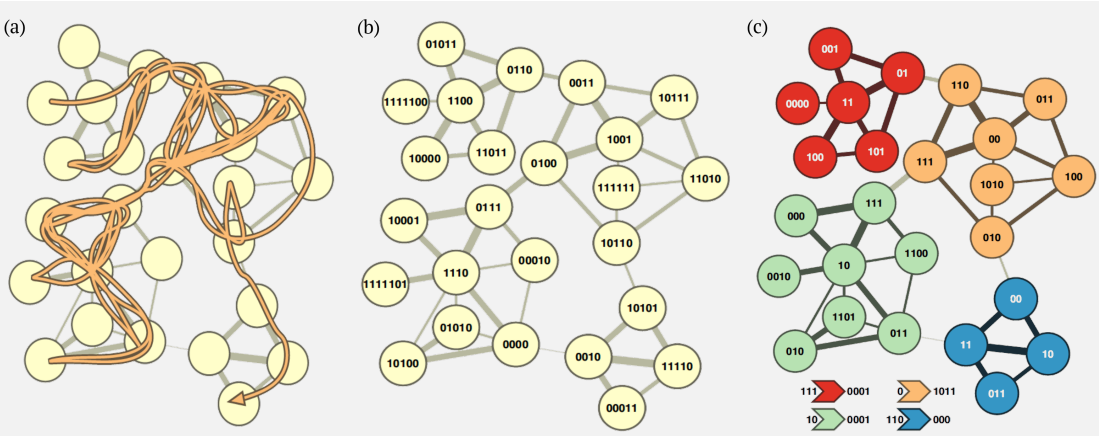

Figure 2.2. ***Main steps of Infomap.*** *The Infomap algorithm detects communities by compressing the description of information flows on networks. First, a random walker (orange trajectory) explores the system (a) and a unique code is given to every node in the network based on the number of times the walker has visited the node, i.e. the Huffman code (b). Then, such a code is compressed to obtain a coarse-grained description of the entire network (c). Adapted from [200].*

theoretic considerations addressing the following question: what is the most parsimonious way to describe an infinitely long random walk on a graph? [202]. As seen in the previous chapter, the information content of any description is provided in terms of the total amount of bits required to describe the different steps (i.e. the description length) [204]. The simplest way to represent the random walk on a graph consists in sequentially reporting all the nodes visited by the walker, where distinct codewords unequivocally identify nodes. However, if the network has a community structure, there might exist more compact representations. One of these representations mimics the principles of of geografic maps, as mentioned in the original paper [202], or postcodes. For instance, most of the Italian cities/regions have unique names, but street names are reused from one city to the next, such that each city has a "Via della Libertá", "Via Cavour", "Via Garibaldi" and so forth. In other words, within the same country[1] it is often the case that large-scale objects, such as cities, can be identified by a unique name, while fine-grain details, as streets, might have the same name across cities. The reuse of street names within a country rarely causes confusion, because most routes remain within the bounds of a single city [202]. Hence, in

[1]Notice that in the case of Italian cities/regions, such multi-level distinction is present, while it may not hold for some other countries.

the case of a network, each community is identified by a unique codeword (the city name in the city metaphor), while the ID of nodes could be reused among different communities (the streets in the city methapor). In other words, nodes with identical names are therefore distinguished by specifying the community they belong to (see Fig. 2.2). The two-level representation just described allows to create a map, which exploits the information-theoretic duality between (i) the minimisation of the description length of a random walker's movements on a network and (ii) the identification of community structure in the network. To provide a concrete example, let us consider a network where clusters are strongly separated from one another. In this case, transitions of a random walker between communities are rare. As a result, it is advantageous to describe the random walker's movements using a two-level description since the clusters' codewords will not be repeated multiple times, while there is a significant saving in the description due to the limited length of the codewords used to denote nodes [199]. The map equation defined in Refs. [201, 202] is the description length of an infinite random walk, and it consists of two terms representing the Shannon entropy of the movement between and within clusters. To obtain a community partition of a graph, Infomap finds the (best) partition yielding the minimum description length.

In the last few years, we have also witnessed a number of studies coming from the field of algorithmic information theory, which provide new approaches for studying the complexity of many complex systems. As mentioned in the previous section, algorithmic complexity, unlike entropy measures, remains unvaried under different descriptions of the same object. That feature is extremely useful when characterising mathematical properties of graphs and networks [144–146, 210]. For example, using approximations of the Kolmogorov complexity, we can detect topological properties related to symmetry, connectedness, and density of a graph [210]. For instance, approximations of the Kolmogorov complexity have been used to track the emergence of the giant component in the ER random graph model [146], or to distinguish graphs generated from different network models [210]. Preliminary results have also shown that it is possible to

check the planarity of a graph by looking at the algorithmic complexity of that graph [146]. Some of these algorithmic information considerations will be used later on in the thesis to justify and present a novel measure of complexity for multiplex systems.

2.2 Multiplex networks

Let us consider the general case of a multiplex network $\mathcal{M}$ made of N nodes interacting through M layers. The network can be formally described by a set of adjacency matrices [36, 38]:

$$\mathbf{A} = \left\{ A^{[1]}, A^{[2]}, \ldots, A^{[M]} \right\} = \left\{ a_{ij}^{[\alpha]} \right\}$$

where the entry $a_{ij}^{[\alpha]}$ is equal to 1 if i and j are connected by a link at layer α, while $a_{ij}^{[\alpha]} = 0$ otherwise. Hereafter, we will use Greek superscripts to denote the layer index, while Roman subscripts will be used for node indexes. When connections among nodes are weighted, the mathematical representation relies on a set of weighted adjacency matrices $\mathbf{W} = \left\{ W^{[1]}, W^{[1]}, \ldots, A^{[M]} \right\} = \left\{ w_{ij}^{[\alpha]} \right\}$, where the weight of the connection between node i and node j at layer α is expressed by $w_{ij}^{[\alpha]}$ [211]. It is worth mentioning that within this formulation, replicas of the same nodes across layers are identified with the same unit, i.e. node i at layer α and node i at layer β represent the same entity and not two different nodes. For instance, social systems can be modelled by means of a multiplex framework. Here, individuals (i.e. nodes) can interact using different communication channels (i.e. layers), including face-to-face interactions, phone calls, e-mail, Instagram, Facebook, Twitter and the like. And yet, the replica nodes represent the same individual across all the different communication channels. In network theory, edges belonging to each layer are often referred to as *intra-links*. The multiplex structure with intra-links can also be found in the mathematical literature under the name of *edge-coloured graphs*.

However, when it comes to modelling certain real-world systems - as in the case of airline

or urban transportation networks [38, 56, 57] - it is often appropriate to take into account the cost of switching modes between the same replica nodes. For instance, when analysing urban transportation networks, it can take a certain amount of time - or even a monetary cost - to transit between bus and rail transportation at a given location [57]. In the airline transportation network, different companies can use two different terminals within the same airport, thus introducing a delay [56]. Hence, considering inter-layer links, or *inter-links*, comes in handy when we are interested in accurately describing these systems. Formally, for every node i and pair of layers α and β, a coupling matrix $C_i = \left\{c_i^{[\alpha,\beta]}\right\}$ is often introduced to account for the inter-links between layers [212]. In general, the element $c_i^{[\alpha,\beta]}$ is a real number, representing the cost of switching layer from α to β for node i. Consequently, a multiplex network with both intra-links and inter-links can be represented by the two vectors of matrices $\mathbf{A}$ and $\mathbf{C}$, or more precisely as [212]:

$$\mathcal{M} = \{\mathbf{A}, \mathbf{C}\} = \left\{A^{[1]}, \ldots, A^{[M]}, C_1, \ldots, C_N\right\}. \tag{2.7}$$

Alternatively, if all replica nodes are coupled with the same intensity c, a common representation is given by the $N \cdot M \times N \cdot M$ supra-adjacency matrix $\mathcal{A}_S$, whose block structure is given by [36, 38]:

$$\mathcal{A}_S = \left[\begin{array}{c|c|c|c} A^{[1]} & c\,I & \ldots & c\,I \\ \hline c\,I & A^{[2]} & \ldots & c\,I \\ \hline \vdots & \vdots & \ddots & \vdots \\ \hline c\,I & c\,I & \ddots & A^{[M]} \end{array}\right]$$

where I represents the $N \times N$ identity matrix. This representation, however, is mostly examined when studying dynamical processes on multiplex networks, such as diffusion [40] or spreading [21, 213–215]. The main idea here is to compare the cost to navigate between nodes of the same layer with the cost of shifting between layers. In this way, there might be an incentive to move on the same layer (e.g. rail, metro or bus) compared to switching mediums [38].

Nonetheless, when it comes to storing the information of a multiplex network, the block structure of the supra-adjacency matrix is redundant. Hence, an edge-list representation based on the vector of adjacency matrices $\left\{A^{[1]}, \ldots, A^{[M]}\right\}$ is often preferred. In the general case, this consists of a list of quadruples $\left(\alpha, i, j, w_{ij}^{[\alpha]}\right)$ representing the strength of the connection $w_{ij}^{[\alpha]}$ between node i and node j at layer α.

Notice that it is always possible to compress the information contained in a multiplex network into single-layer structures. For instance, the *aggregate network* or *projected network* $\mathcal{A}$ [38, 216] represents the (binary) network obtained by disregarding the layered structure of the multiplex network, defined as:

$$\mathcal{A} = \{a_{ij}\} = \begin{cases} 1 & \text{if } \exists\, \alpha : a_{ij}^{[\alpha]} = 1 \\ 0 & \text{otherwise.} \end{cases} \tag{2.8}$$

In other terms, two nodes i and j are connected in the aggregate network if and only if they share at least one connection in the multiplex. See Fig. 2.3 for a pictorial representation of a multiplex network and the corresponding aggregate network. A more valuable single-layer description in terms of the information retained from the multiplex relies on the *overlapping matrix* $\mathcal{O}$ [216]. This matrix is defined as $\mathcal{O} = \{o_{ij}\} = \sum_\alpha a_{ij}^{[\alpha]}$, such that each entry accounts for the total number of connections between a pair of nodes across all the layers.

The aggregate and overlapping matrix are both often used as a benchmark to identify genuine multiplex phenomena emerging from the multiplexity characterising interconnected systems [38, 56]. Indeed, as we will show throughout this thesis, in many circumstances the single-layer representation - based either on $\mathcal{A}$ or $\mathcal{O}$ - is not adequate to fully characterise the multilayer structure. Another important class of systems that can be described by the multiplex framework [i.e. Eq. 2.7] is that of temporal networks. Here, links generally change over time, so that connections may appear and disappear in time. Remarkably, thanks to the unprecedented availability

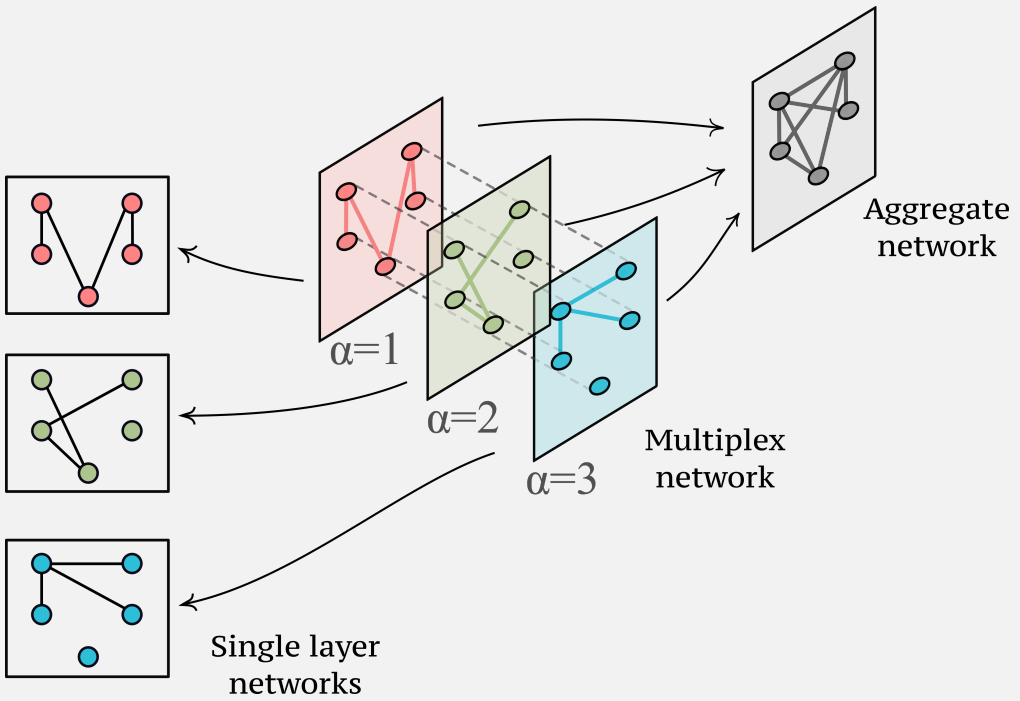

Figure 2.3. ***Schematic representation of a multiplex network with inter-links***. *Multiplex network with $M = 3$ layers and $N = 5$ nodes. Solid lines represent the intralayer links, while the dashed lines are the interlayer links (all the nodes are connected with their counterparts in each layer). The multiplex structure usually conveys more information than the collection of the layers taken in isolation (left panel) and of the aggregate network (right panel).*

of high-resolution longitudinal data sets, it is now possible to accurately examine the evolution of many biological, social, and human networked systems. In the context of social dynamics, for instance, SocioPattern data [217] and the Copenhagen Networks Study [218] represent two exceptional examples of high-quality temporal data. In these data sets, physical proximity sensors (or Bluetooth strength signals) provide information about face-to-face interactions between individuals, while smartphones record the activity of text messages, phone calls communications, or even information regarding Facebook friendships. For all these temporal events, interactions might occur instantaneously or last for a certain amount of time (e.g. face-to-face interactions and phone calls). For this reason, when it comes to modelling, temporal networks are often represented using a sequence of adjacency matrices $\left(A^{[1]}, \ldots, A^{[\tau]}, \ldots A^{[\mathcal{T}]}\right)$, where each $A^{[\tau]}$ describes the network of interaction observed at 'time' τ. In particular, each layer encodes instantaneous connections or, more generally, interactions that occurred over a specific time window of length

$\delta\tau$.

Notice that the temporal ordering between layers in this context is essential. When representing time-varying systems as multiplex networks, the matrices $A^{[\tau]}$ and $A^{[\tau+1]}$ represent the connections of the time-varying networks at two consecutive time measurements. Furthermore, when we are interested in considering inter-links between subsequent layers, the coupling matrix C_i assumes a particular form. That is, for each node i the matrix C_i assumes the following form: $C_i = c_i^{[\alpha,\beta]} = \delta_{\alpha+1,\beta}$, where $\delta_{\alpha,\beta}$ represents the Kronecker delta. Such a particular form of C_i is due to the fact that node i at layer τ is only linked to the same node at layer $\tau + 1$, i.e. replica nodes are only linked forward in time. For a formal discussion and detailed analysis of temporal networks, see Refs. [16–18, 219, 220].

A last point worth noting is that the vectorial description introduced for the multiplex network, along with the supra-adjacency matrix representation, can be framed within a more general tensorial formalism, first introduced in Ref. [221]. Here, all the possible connections between and across layers are encoded in a rank four (2-covariant, 2-contravariant) tensor $T = \left\{t_{i,j}^{[\alpha,\beta]}\right\}$. In particular, the entry $t_{i,j}^{[\alpha,\beta]}$ generally represents the (possibly weighted) interaction between node i at layer α and node j at layer β. Hence, the tensor T perfectly describes interconnected and interdependent connections within the general multilayer framework. For instance, the supra-adjacency matrix $\mathcal{A}_S$ corresponds to the "flattening" of the tensor T. The tensor T describes a multiplex network with inter-links only when, for $\alpha \neq \beta$, the elements corresponding to the entries $t_{i,j}^{[\alpha,\beta]}$ are equal to zero with $i \neq j$. A rigorous classification of multiplex networks, and more generally multilayer networks, can be found in Refs. [37, 221]. However, notice that throughout this thesis, we will not use the tensorial representation. Instead, we will rely on the multiplex network formalism (i.e. the one based on the vectorial formulation) for the intrinsic simplicity in describing multilayer systems.

2.3 Structural measures of multiplex networks

When adopting a multiplex framework, many of the structural metrics proposed for single-layer networks must be adapted to properly take into account the distinct relationships within the multiplex structure. As a matter of fact, in the traditional single-layer approach, a simple scalar variable describes node features of a network, in stark contrast to the situation for multiplex systems, where node properties are generally encoded into vectorial variables. In this section, we provide an overview of the salient structural metrics recently proposed in the context of multiplex networks. We begin by introducing some of the local and global metrics associated with nodes and edges. We then present some of the layer-based metrics and, finally, we examine the main measures introduced to quantify the amount of correlations in a multiplex.

2.3.1 Node properties

Perhaps one of the simplest yet most effective measures to quantify the importance of a node in a network is based on its degree. For an unweighted and undirected multiplex network $\mathcal{M}$ with M layers and N nodes, we define the *multiplex degree* [38, 216] as $\mathbf{k}_i = \left(k_i^{[1]}, k_i^{[2]}, \ldots, k_i^{[M]}\right)$, where each element $k_i^{[\alpha]}$ represents the degree of node i at layer $\alpha = 1, \ldots, M$ defined as:

$$k_i^{[\alpha]} = \sum_j^N a_{ij}^{[\alpha]}. \tag{2.9}$$

The multiplex degree $\mathbf{k}_i$ provides insights into how the number of connections are distributed across the layers around node i. Moreover, by examining this quantity for a specific node i and by comparing it with all the other nodes, we can have a glance on how important is that node across the layers. Indeed, it is often the case that certain nodes have a lot of connections in only a few layers, while being isolated or holding few connections in the remaining layers. A compact structural measure is obtained when considering the total number of connections of node i. This

is known as the *total degree* or *overlapping degree* [38, 216], and it is defined as:

$$o_i = \sum_{\alpha=1}^{M} k_i^{[\alpha]}. \tag{2.10}$$

In other words, this measure accounts for the number of links incident on i across all the layers. By analysing the distribution of the total degree, we can have a rough ranking related to the "importance" of nodes in the multiplex in terms of the number of incident edges. Unsurprisingly, a node with high number of connections is likely to have an important function in the multiplex. However, it is often crucial to characterise the importance of nodes across the layers, thus distinguishing nodes that appear to be hubs in single layers from nodes that are intrinsically multiplex (i.e. they have a certain amount of connections heterogeneously distributed across the layers). To quantify how heterogeneously the connections around nodes are distributed, the authors of Ref. [216] introduced the *multiplex participation coefficient*, defined as:

$$P_i = \frac{M}{M-1}\left[1 - \sum_{\alpha=1}^{M}\left(\frac{k_i^{[\alpha]}}{o_i}\right)^2\right]. \tag{2.11}$$

The multiplex participation coefficient P_i takes values in the interval $[0, 1]$ thanks to the normalisation factor $\frac{M}{M-1}$ (in the limit case when $\frac{k_i^{[\alpha]}}{o_i}$ is smallest for all α, i.e. it is equal to $\frac{1}{M}$, the term inside the square brackets becomes $1 - \frac{1}{M}$). In particular, $P_i = 0$ when a node is active in only one layer, while $P_i = 1$ when the links around node i are equally distributed across the layers. In Ref. [216] the authors analysed several real-world multiplex networks, observing that the rank distribution of P_i can be used to provide useful insights regarding the role of nodes in the multiplex. From this observation, they propose a way to classify nodes of multiplex systems by simultaneously looking at the z-score of the total degree o_i and at the multiplex participation coefficient P_i. In other words, they created a cartography of nodes (in the spirit of Refs. [222, 223]), distinguishing nodes that are properly multiplex from nodes that are mainly active on single layers. More precisely, they introduce the z-score associated with the total degree

in order to remove the dependence of the multiplex size as:

$$z(o_i) = \frac{o_i - \langle o \rangle}{\sigma_o} \tag{2.12}$$

where $\langle o \rangle$ and σ_o represent the mean and standard deviation of the total degree o_i respectively. They then classify nodes in two different classes according to the values of the z-score, namely, *hubs* [for which $z(o_i) \geq 2$], and *regular nodes* [for which $z(o_i) < 2$]. They distinguish three classes when considering the values of the participation coefficient P_i as following:

$$\begin{cases} \text{if } 0 \leq P_i \leq \frac{1}{3} & \text{focused nodes} \\ \text{if } \frac{1}{3} < P_i \leq \frac{2}{3} & \text{mixed nodes} \\ \text{if } P_i > \frac{2}{3} & \text{truly multiplex nodes.} \end{cases} \tag{2.13}$$

As a consequence, the nodes of a multiplex can be mapped into six different classes of nodes in the plane $[P_i, z(o_i)]$, leading to the first example of multiplex cartography in the literature. As an example, we report in Figure 2.4 the cartography obtained for the real-world multiplex system of Top Noordin Indonesian Terrorists (where $M = 3$) [216]. In particular, from this cartography, it is evident that certain nodes may have the same total number of links, yet have completely different roles in the multiplex [highlighted in Fig. 2.4(b)]. Notice that to quantify the heterogeneity of the distribution of the multiplex degree $\mathbf{k}_i$, an information-theoretic measure based on Shannon entropy can be likewise used [216]:

$$H_i = -\sum_{\alpha=1}^{M} \frac{k_i^{[\alpha]}}{o_i} \log\left(\frac{k_i^{[\alpha]}}{o_i}\right). \tag{2.14}$$

When analysing real-world multiplex networks, it is common to find nodes that are active only in certain layers of the multiplex (i.e. their degrees are equal to zero in some layers). Based on this observation, in Ref. [224] the authors introduced the *node activity vector*, which describes

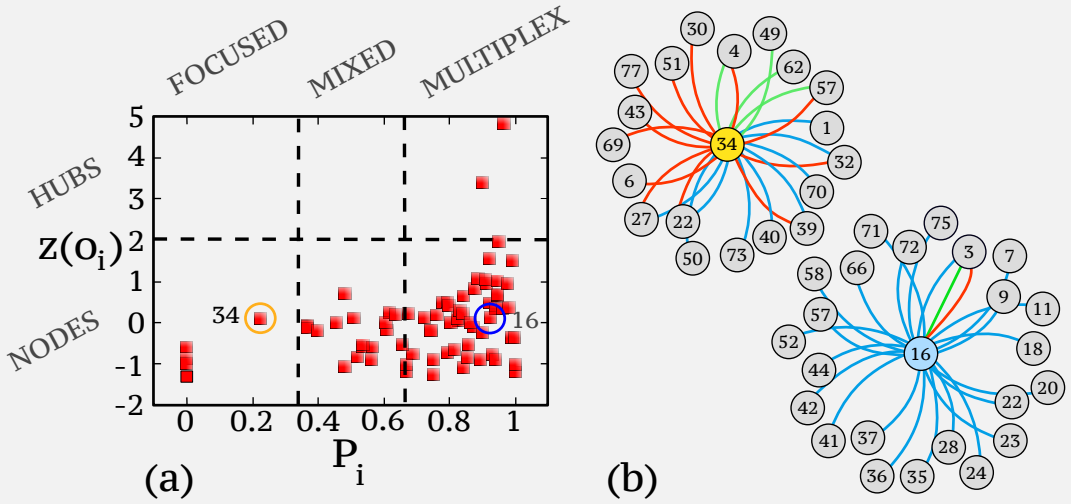

Figure 2.4. ***Multiplex carthography of a real-world multiplex system***. *(a) Roles of the nodes of the Top Noordin Indonesian Terrorists multiplex network (M=3). The layers considered correspond to trust, operational, and communication. In particular, by mapping nodes in the plane* $[P_i, z(o_i)]$*, we can classify nodes according to their roles in the multiplex. For instance, despite nodes* 34 *and* 16 *have the same value of* $z(o_i)$*, they appear to have very different roles in the multiplex (reflected by the different values of the multiplex participation coefficient* P_i*). (b) The ego networks of the two nodes clearly show the connectivity patterns across layers. While the links of node* 16 *are homogeneously distributed in the multiplex, the edges around node* 34 *appears to be focused only in just one layer (blue links, operational layer). Adapted from [216].*

the pattern of connection in the different layers or, in short:

$$\mathbf{b}_i = \left\{ b_i^{[1]}, \ldots, b_i^{[M]} \right\} \tag{2.15}$$

where $b_i^{[\alpha]} = 1 - \delta_{0,k_i^{[\alpha]}}$, so that $b_i^{[\alpha]} = 1$ if node i is active on layer α, and zero otherwise. The *node activity* B_i of node i is then defined as the number of times the node i is active across the layers of the multiplex:

$$B_i = \sum_{\alpha=1}^{M} b_i^{[\alpha]} \tag{2.16}$$

so that, by definition, we have $o \leq B_i \leq M$. We will study this quantity later in the thesis when analysing and modelling multilayer transportation networks.

As seen in Section 2.1, shortest paths also play an important role in efficiently navigating single-layer networks. A similar characteristic is present in the context of multiplex networks. This

is because the layered structure of a multiplex network influences the properties of the paths in critical ways. For instance, in a multiplex network, a path can span more than one layer and, as a result, the shortest paths on single layers are not always informative. To make the matter more complicated, the number of shortest paths between any couple of nodes exponentially increases as a function of the number of layers of a multiplex [216]. Moreover, if we construct a multiplex network with M layers starting from a single isolated layer, the inclusion of more layers can only decrease the shortest path distance between any pair of two nodes [38]. For this reason, in a multiplex framework, nodes might be reached from any other location of the network in very few steps, compared to the corresponding single-layer aggregate. To capture the effect that the multiplex representation has on the reachability of each node of the system, the *node interdependence* has been introduced [225]. This quantity is defined as:

$$\lambda_i = \frac{1}{N-1} \sum_{\substack{j \in N \\ j \neq i}} \frac{\psi_{ij}}{\sigma_{ij}}. \tag{2.17}$$

In the expression, ψ_{ij} is the number of shortest paths between i and j that span across more than one layer, while σ_{ij} is the total number of shortest paths between i and j in the multiplex. If $\lambda_i \approx 1$ then i fully exploits the multiplex structure of the system to reach other nodes, while if $\lambda_i \approx 0$ node i reaches other nodes through shortest paths whose edges are on just one layer. By averaging λ_i over all the nodes, we obtain the so-called *interdependence*, i.e.:

$$\lambda = \frac{1}{N} \sum_{i=1}^{N} \lambda_i. \tag{2.18}$$

Finally, the authors of Ref. [226] introduced the *node pairwise multiplexity* Q_{ij}, which is defined as the fraction of layers where node i and j are simultaneously active, i.e.:

$$Q_{i,j} = \frac{1}{M} \sum_{\alpha=1}^{M} b_i^{[\alpha]} b_j^{[\alpha]}. \tag{2.19}$$

2.3.2 Edge properties

One of the crucial signatures associated with multiplex networks is related to the configuration of links across layers. As a matter of fact, in a multiplex network the same pair of nodes can be connected through several edges across layers in a number of different ways. In a general multiplex with M layers, the amount of edge overlap between any pair of nodes can be encoded in the *overlapping matrix* O, where the generic element is defined as [36, 216]:

$$o_{ij} = \frac{1}{M} \sum_{\alpha=1}^{M} a_{ij}^{[\alpha]}. \tag{2.20}$$

The measure can be extended to the whole multiplex by averaging over all the pairs of nodes that share at least one edge in common [227]:

$$\omega = \frac{\sum_{\alpha} \sum_{i} \sum_{j>i} a_{ij}^{[\alpha]}}{M \sum_{i} \sum_{j} (1 - \delta_{0,o_{ij}})}. \tag{2.21}$$

Alternative definitions charactering the overlap of links in a multiplex network were also considered in the literature [36, 228, 229]. For example, the *local overlap* (with a slight different notation from the original paper) of the links in layer α and layer β counts the number of overlapping edges that are incident to node i in both layers:

$$\nu_i^{[\alpha\,\beta]} = \sum_{j=1}^{N} a_{ij}^{[\alpha]} a_{ij}^{[\beta]}. \tag{2.22}$$

In the case of an undirected network, the local overlap quantifies the similarity between the connection patterns of node i in layer α and β. The edge overlap plays a crucial role in many real-world systems and, as we shall see in this chapter and the next, real-world multiplex systems exhibit a non-negligible amount of overlap.

However, most of the synthetic measures of overlap introduced above provide only a count of the number of links in overlap, but neglecting the pattern of connections across layers. One possible way to keep track of the pattern of all the connections is using the so-called *multilinks* [58, 230]. For a multiplex network with M layers, let us consider the following vector [38]:

$$\mathbf{m} = \left(m^{[1]}, \ldots, m^{[\alpha]}, \ldots, m^{[M]}\right) \tag{2.23}$$

where $m^{[\alpha]} \in \{0, 1\}$. Any pairs of nodes i and j are connected by a multilink $\mathbf{m}$ if and only if they are connected in every layer α in which $m^{[\alpha]} = 1$, and they are disconnected in every layer α in which $m^{[\alpha]} = 0$ [38]. In other words, the vector $\mathbf{m}$ represents the pattern of connections in a multiplex with M layers. In the case of an unweighted multiplex, every pair of nodes i and j is connected by a multilink $\mathbf{m}_{ij}$ of the form [38, 211]:

$$\mathbf{m}_{ij} = \left(a_{ij}^{[1]}, \ldots, a_{ij}^{[M]}\right) \tag{2.24}$$

so that $\mathbf{m}_{ij}$ represents the pattern of connection of the edge (i, j) across all the M layers. Notice that whenever node i and node j are connected in at least one layer, i.e. $\mathbf{m}_{ij} \neq \mathbf{0}$, we say that they are connected by a non-trivial multilink. Multilinks are mutually exclusive, i.e., any pair of nodes (i, j) can be linked only by one multilink $\mathbf{m}_{ij}$. Clearly, given a multiplex with M layers, the total number of non-trivial multilinks is $2^M - 1$. Figure 2.5 portrays a schematic illustration of the multilinks for a multiplex system with $M = 2$ layers. In addition, with the concept of multilink in mind, we can define the multidegree $k_i^{[\mathbf{m}]}$ of node i as the total number of multilinks $\mathbf{m}$ incident to node i. Formally, this is defined as [38, 230]:

$$k_i^{\mathbf{m}} = \sum_{j \neq i} \delta_{\mathbf{m}, \mathbf{m}_{ij}} \tag{2.25}$$

where we have that $\delta_{\mathbf{m}, \mathbf{m}_{ij}} = 1$ for $\mathbf{m} = \mathbf{m}_{ij}$, and $\delta_{\mathbf{m}, \mathbf{m}_{ij}} = 0$ otherwise.

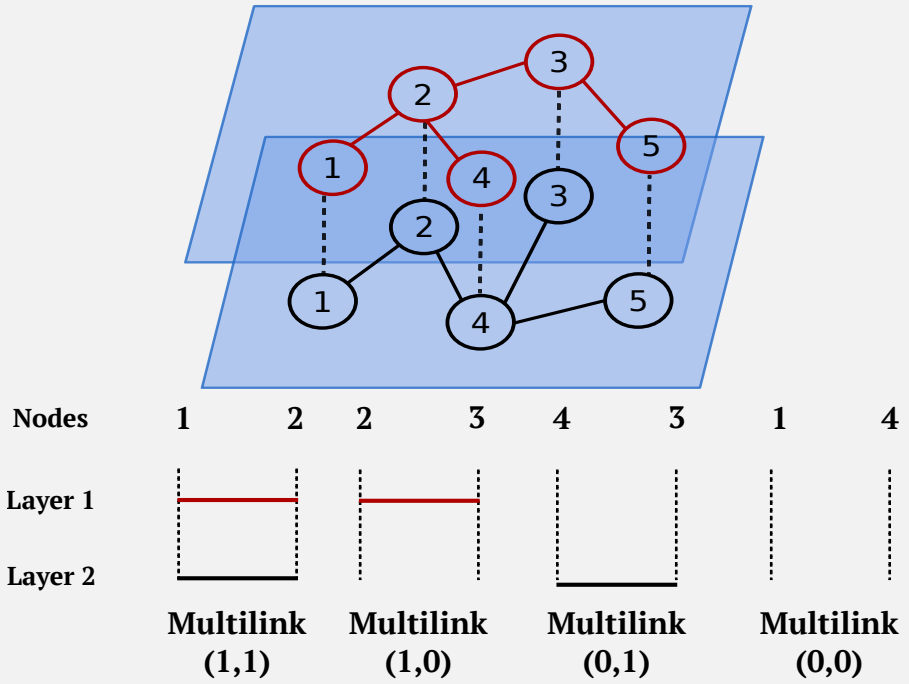

Figure 2.5. ***Schematic illustration of multilinks in a multiplex network.*** *We report for a multiplex with $M = 2$ layers the graphical representation of all possible multilinks of the form $\mathbf{m} = (m^{[1]}, m^{[2]})$. Reprinted from [211].*

2.3.3 Layer properties

As was the case with node activity B_i, it is possible to define the activity of a given layer α by considering the *layer-activity vector*:

$$\mathbf{d}^{[\alpha]} = \left\{ b_1^{[\alpha]}, b_2^{[\alpha]}, \ldots, b_N^{[\alpha]} \right\} \tag{2.26}$$

where, again $b_i^{[\alpha]} = 1$ if node i is active at layer α, i.e. $k_i^{[\alpha]} > 0$, and zero otherwise. We therefore define the *layer activity* of layer α [224] as the number $N^{[\alpha]}$ of active nodes in α, which is equal to the number of nonzero elements of $\mathbf{d}^{[\alpha]}$:

$$N^{[\alpha]} = \sum_i b_i^{[\alpha]}. \tag{2.27}$$

By definition, we have $0 \leq N^{[\alpha]} \leq N$. We can now introduce some measures to characterise the similarities between layer activities in a multiplex network. For instance, the *pairwise multiplexity* $Q_{\alpha,\beta}$ of two layers α and β is defined as [224]:

$$Q_{\alpha,\beta} = \frac{1}{N}\sum_i b_i^{[\alpha]} b_i^{[\beta]}. \tag{2.28}$$

In other words, this quantity corresponds to the fraction of nodes that are simultaneously active in both layers α and β, and it is such that $Q_{\alpha,\beta} \in [0,1]$. Clearly, the higher the pairwise multiplexity of the two layers, the more similar the activity pattern of the nodes across those layers.

In order to quantify the relative overlap between two layers at the level of node activity, the authors of Ref. [224] introduced the *normalized Hamming distance*, defined as:

$$H_{\alpha,\beta} = \frac{\sum_i b_i^{[\alpha]}(1-b_i^{[\beta]}) + (1-b_i^{[\alpha]})b_i^{[\beta]}}{\min(N^{[\alpha]}+N^{[\beta]}, N)}. \tag{2.29}$$

In short, $H_{\alpha,\beta}$ corresponds to the total number of differences in the activities of layers α and β divided by the maximum possible number of those differences. In particular, when $\mathbf{d}^{[\alpha]} = \mathbf{d}^{[\beta]}$ follows that $H_{\alpha,\beta} = 0$, while $H_{\alpha,\beta} = 1$ when all the active nodes at layer α are not active in the other layer, and vice versa.

2.3.4 Inter-layer degree correlation

After numerous studies in the network science community, it is now widely known that real-world single-layer networks are characterised by the presence of non-trivial degree correlations [231–235]. For example, social and information networks often exhibit *assortative degree correlation*, meaning that nodes with a certain degree are usually connected with nodes having a similar degree. By contrast, biological and technological networks are characterised by *disassortative*

degree correlation, so that nodes of a low degree are preferentially attached with nodes having a high degree, or vice versa. However, in a multilayer formulation, beside the classical concept of intra-layer degree-degree correlations, the presence of correlation might appear across layers. For instance, it is often interesting to look at how a certain node property at a given layer is correlated with the same - or even other - node properties at another layer. Here, we shall follow the discussion presented in Ref. [38], trying to introduce the interlayer degree correlation measures that have been proposed in the context of multilayer networks, highlighting the pros and cons of each of these metrics.

Let us start with a multiplex network consisting of two layers α and β. We can define the probability of finding a node of degree $k^{[\alpha]}$ in layer α and with degree $k^{[\beta]}$ in layer β as [38, 224]

$$P\left(k^{[\alpha]}, k^{[\beta]}\right) = \frac{\mathcal{N}\left(k^{[\alpha]}, k^{[\beta]}\right)}{N} \tag{2.30}$$

where $\mathcal{N}\left(k^{[\alpha]}, k^{[\beta]}\right)$ represents the number of nodes with degree $k^{[\alpha]}$ in layer α and with degree $k^{[\beta]}$ in layer β. By definition, the presence of interlayer degree correlation in a multiplex network can be spotted when Eq. (2.30) does not factorise into the product of the probabilities, i.e. $P\left(k^{[\alpha]}, k^{[\beta]}\right) \neq P\left(k^{[\alpha]}\right) P\left(k^{[\beta]}\right)$. However, to quantitatively measure the interlayer degree correlations, the authors of Ref. [227] considered the pointwise mutual information between the degree sequences of the two layers, defined as:

$$I^{[\alpha,\beta]} = \sum_{k^{[\alpha]},k^{[\beta]}} P\left(k^{[\alpha]}, k^{[\beta]}\right) \log \frac{P\left(k^{[\alpha]}, k^{[\beta]}\right)}{P\left(k^{[\alpha]}\right) P\left(k^{[\beta]}\right)}. \tag{2.31}$$

With this definition, $I^{[\alpha,\beta]} = 0$ when the degree sequences of the two layers are uncorrelated, while the larger $I^{[\alpha,\beta]}$, the higher the correlation of the two degree sequences. However, depending on the size of the network, this quantity might become a less solid statistic, so that other correlation measures are usually preferred [38]. In particular, one of the alternative approaches is analogous to the one used to detect degree correlation in single-layer network [i.e. the one based on degree

correlation function $k_{nn}(k)$]. More precisely, we define the average degree of a node at layer α conditioned on the degree of the same node in layer β as [236, 237]:

$$\left\langle k^{[\alpha]}|k^{[\beta]}\right\rangle = \sum_{k^{[\alpha]}} k^{[\alpha]} P\left(k^{[\alpha]}|k^{[\beta]}\right) = \frac{\sum_{k^{[\alpha]}} k^{[\alpha]} P(k^{[\alpha]}, k^{[\beta]})}{\sum_{k^{[\alpha]}} P(k^{[\alpha]}, k^{[\beta]})} \tag{2.32}$$

where $P\left(k^{[\alpha]}|k^{[\beta]}\right)$ denotes the conditional probability distribution, which corresponds to the probability of finding a node of degree $k^{[\alpha]}$ at layer α given that the same node has degree $k^{[\beta]}$ at layer β. Notice also that all the summations run over the degree classes. If $\left\langle k^{[\alpha]}|k^{[\beta]}\right\rangle$ does not depend on $k^{[\beta]}$, then the degree sequences of the two layers are uncorrelated. By contrast, an increase (or decrease) of $\left\langle k^{[\alpha]}|k^{[\beta]}\right\rangle$ as a function of $k^{[\beta]}$ reflects the presence of assortative (or disassortative) interlayer degree correlations between the two degree sequences. However, when such a function does not have a monotonic increase (or decrease), the interpretation of the results might become ambiguous.

Finally, we shall present three different coarse-grained correlations metrics based on the Pearson, Spearman, and Kendall correlation coefficients respectively. First, to quantitatively measure the presence of interlayer degree correlation in a multiplex network by relying on the Pearson's correlation coefficient we can consider the following quantity [224]:

$$r_{\alpha,\beta} = \frac{\langle k_i^{[\alpha]}, k_i^{[\beta]}\rangle - \langle k_i^{[\alpha]}\rangle\langle k_i^{[\beta]}\rangle}{\sigma_{k^{[\alpha]}}\sigma_{k^{[\beta]}}} \tag{2.33}$$

where $\langle k_i^{[\alpha]}\rangle$ and $\sigma_{k^{[\alpha]}}$ represent the mean and standard deviation of the multiplex degree sequence. Notice that for the computation of the correlation coefficient, only the nodes active on both layers are considered. Moreover, if the two degree distributions are broad, the Pearson's correlation coefficient might be strongly influenced by the correlation of the high-degree nodes [38]. Indeed, in this case, a large value of correlation coefficient does not always imply a high linear relationship between the two degree distribution.

Another coarse-grained measure was presented in Ref. [224], where the authors quantify inter-

layer degree correlations ρ by using the Spearman's rank correlation coefficient. This is defined as:

$$\rho = \frac{\sum_i (R_i^{[\alpha]} - \overline{R^{[\alpha]}})(R_i^{[\beta]} - \overline{R^{[\beta]}})}{\sqrt{\sum_i (R_i^{[\alpha]} - \overline{R^{[\alpha]}})^2 \sum_j (R_j^{[\beta]} - \overline{R^{[\beta]}})^2}} \tag{2.34}$$

where $R_i^{[\alpha]}$ represents the rank induced by the degree of node i at layer α, while $\overline{R^{[\alpha]}}$ is the average rank of nodes at layer α. Also in this case, the quantity is computed only considering nodes that are active in both layers. However, such a measure has some degree of degeneracy if ties (i.e. some nodes exhibit the same degree) are present in the degree sequences.

A measure that takes into account the existence of ties in the node ranks is obtained by using the Kendall's correlation coefficient. This is defined as [224, 237]:

$$\tau_{\alpha,\beta} = \frac{n_c^{[\alpha,\beta]} - n_d^{[\alpha,\beta]}}{\sqrt{(n_0 - n^{[\alpha]})(n_0 - n^{[\beta]})}}, \tag{2.35}$$

where:

$$n_0 = \frac{1}{2} NQ_{\alpha,\beta}(NQ_{\alpha,\beta} - 1)$$

while the terms $n_c^{[\alpha,\beta]}$ and $n_d^{[\alpha,\beta]}$ represent the number of concordant pairs and the number of discordant pairs in the two rankings respectively. The nodes i and j are considered to be a concordant pair if the ranks of the two nodes in the two layers α and β agree. In other words, if both $R_i^{[\alpha]} > R_j^{[\alpha]}$ and $R_i^{[\beta]} > R_j^{[\beta]}$ or, equivalently, if $R_i^{[\alpha]} < R_j^{[\alpha]}$ and $R_i^{[\beta]} < R_j^{[\beta]}$. Hence, two nodes i and j are a discordant pair, if they are not concordant. Finally, the two quantities $n^{[\alpha]}$ and $n^{[\beta]}$ represent the number of rank ties in the two layers respectively. In particular, when tied values occur, they are each given the average of the ranks that would have been given had no ties occurred.

As we shall see in Chapter 6, both interlayer degree correlation and overlap play a very critical role when analysing the robustness of multiplex systems against targeted attacks. In that context, we will provide a novel algorithm for simultaneously tuning the overlap and the degree

correlations between two layers of a multiplex network.

2.4 Information-theoretic approaches in multiplex networks

A multiplex network generally combines significantly more information than any of its layers or compressed single-layer representations. As an example, if we aim to understand the hidden mechanisms behind the performance of London transportation systems, we need to consider all the information of the different layers of interactions, which includes bus, coach, tube, and rail networks. For this reason, in recent years one of the most crucial challenges in multiplex networks [212] - and more generally in high-order interaction representations [238, 239] - is devising numerical methods and algorithms aimed at extracting relevant features from these complex systems. In this respect, a possible approach relies on exploiting the mesoscale structure, or even the structural correlations, that are observed in multiplex networks. Here, we revise some of the recent contributions in this regard, highlighting the importance of information theory in unveiling hidden structures.

In the context of characterising mesoscale structure, for instance, the authors of Ref. [240] propose an indicator $\tilde{\Theta}^S$ to quantify the similarity between the community structure of two layers[2]. Such an indicator is based on network entropy and information theory tools [183, 187, 241] mentioned in Chapter 2.1. More precisely, by considering a multiplex network $\mathcal{M}$, they consider $q_i^{[\alpha]}$ and $q_i^{[\beta]}$ to be the node community assignment of node i at layer α and β respectively. In this way, such node community assignment induces a block model, where nodes are divided into blocks according to the community label and degree. Using the entropy of network ensembles $\Sigma_{k^{[\alpha]},q^{[\alpha]}}$ introduced in Ref. [230], they are able to evaluate the number of networks with the given block structure. To quantify the significance of the network partition with respect to the network structure, they considered the z-score between the network ensemble $\Sigma_{k^{[\alpha]},q^{[\alpha]}}$ and the

[2]Notice that we shall rely on the nomenclature of "indicator function" for $\tilde{\Theta}^S$, which was introduced in Ref. [240], even though such an indicator can assume real values.

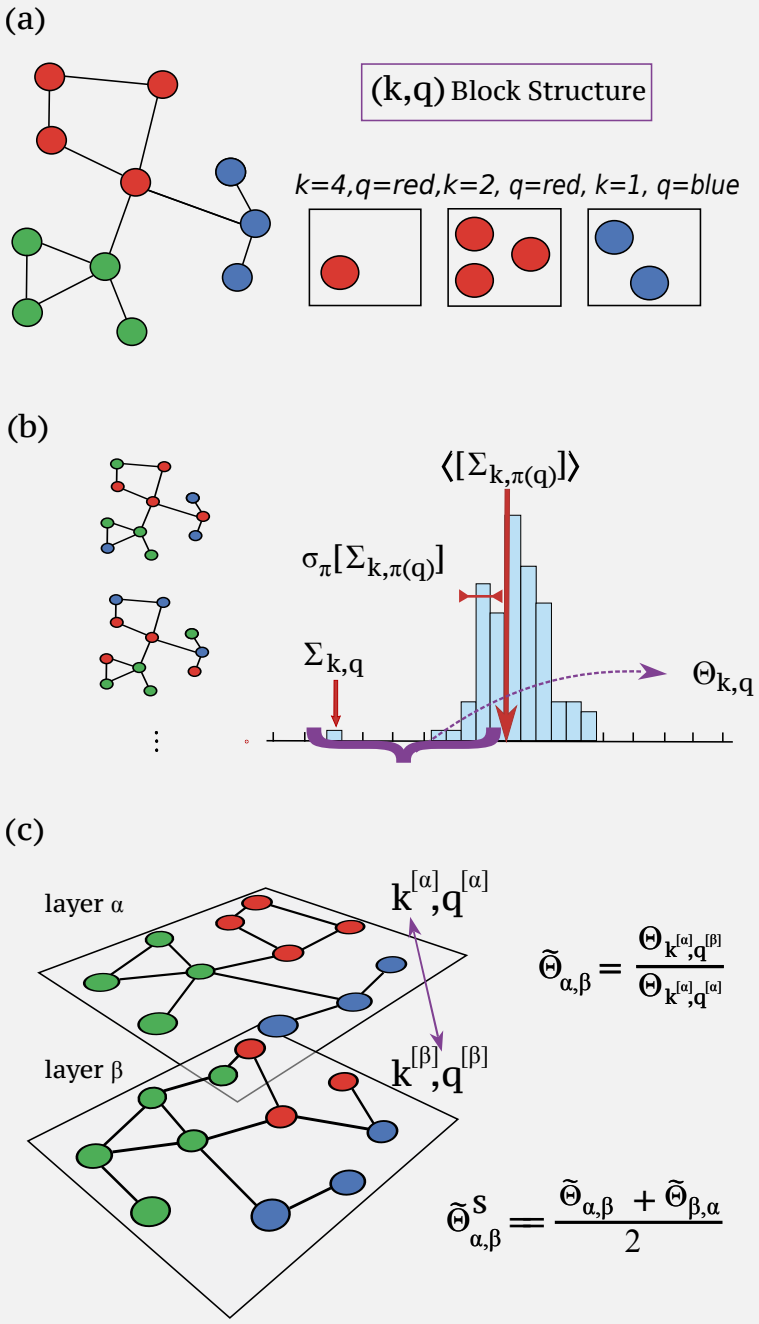

Figure 2.6. ***Pictorial representation of the method for characterising the mesoscale structure of a multiplex network.*** *Firstly, (a) nodes in a layer α are divided into classes $(k^{[\alpha]}, q^{[\alpha]})$ of nodes, where $k^{[\alpha]}$ indicates the node degrees and $q^{[\alpha]}$ the node community assignments on the layer α. These classes induce a block structure specified by the number of connections between the vertices of each class and the number of links connecting the nodes in different classes. Afterwards, (b) the entropy $\Sigma_{k^{[\alpha]},q^{[\alpha]}}$ is computed and compared with the ensemble of randomisation of the node characteristics $\Sigma_{k^{[\alpha]},\pi(q^{[\alpha]})}$. After computing the mean and standard deviation of the entropy distribution, the indicator $\Theta_{k^{[\alpha]},q^{[\alpha]}}$ is calculated. Lastly, (c) given a second layer β, to obtain an unbiased indicator, $\Theta_{k^{[\alpha]},q^{[\beta]}}$ is appropriately normalised to account for the amount of information of the structure in layer α, carried by the node characteristics in layer β, i.e. $\tilde{\Theta}_{k^{[\alpha]},q^{[\beta]}}$. A symmetrised measure is then used $\tilde{\Theta}^{S}_{k^{[\alpha]},q^{[\beta]}}$ quantifies how similar are layer α and β with respect to their community structure. Adapted from [240].*

typical value $\Sigma_{k^{[\alpha]},\pi(q^{[\alpha]})}$ that one obtains when considering a random permutation $\pi(q)$ of the community assignment of the nodes (such an idea was originally proposed in Ref. [242]). This information theory quantity is defined as:

$$\Theta_{k^{[\alpha]},q^{[\alpha]}} = \frac{|\Sigma_{k^{[\alpha]},q^{[\alpha]}} - \langle\Sigma_{k^{[\alpha]},\pi(q^{[\alpha]})}\rangle|}{\sigma_\pi\left[\Sigma_{k^{[\alpha]},\pi(q^{[\alpha]})}\right]} = \frac{|\Sigma_{k^{[\alpha]},q^{[\alpha]}} - \langle\Sigma_{k^{[\alpha]},\pi(q^{[\alpha]})}\rangle|}{\langle\Sigma^2_{k^{[\alpha]},q^{[\alpha]}}\rangle - \langle\Sigma_{k^{[\alpha]},\pi(q^{[\alpha]})}\rangle^2} \tag{2.36}$$

where $\langle\Sigma_{k^{[\alpha]},\pi(q^{[\alpha]})}\rangle$ is the expected value over random uniform permutations $\pi(q^{[\alpha]})$ of the node community assignment $q^{[\alpha]}$ in layer α, while $\sigma_\pi\left[\Sigma_{k^{[\alpha]},\pi(q^{[\alpha]})}\right]$ is the standard deviation of the distribution.

Considering this indicator measure, the authors of Ref. [240] propose to quantify the level of similarity between layers in a multiplex network. Indeed, the indicator $\Theta_{k^{[\alpha]},q^{[\beta]}}$ measures the specificity of the layer α with respect to the particular community assignment $q_i^{[\beta]}$ derived from the mesoscale structure of layer β. Formally, the indicator measure is defined as:

$$\Theta_{k^{[\alpha]},q^{[\beta]}} = \frac{|\Sigma_{k^{[\alpha]},q^{[\beta]}} - \langle\Sigma_{k^{[\alpha]},\pi(q^{[\beta]})}\rangle|}{\sigma_\pi\left[\Sigma_{k^{[\alpha]},\pi(q^{[\beta]})}\right]}.$$

It is worth mentioning that such a quantity is typically normalised to obtain an unbiased estimator $\widetilde{\Theta}_{\alpha,\beta} = \Theta_{k^{[\alpha]},q^{[\beta]}}/\Theta_{k^{[\alpha]},q^{[\alpha]}}$, which is a measure of how the node community assignment in layer β is reflected in layer α (see a pictorial representation of the steps of the method in Fig. 2.6). Finally, the indicator $\widetilde{\Theta}^S_{\alpha,\beta}$ symmetrises $\widetilde{\Theta}_{\alpha,\beta}$, i.e. $\widetilde{\Theta}^S_{\alpha,\beta} = (\widetilde{\Theta}_{\alpha,\beta} + \widetilde{\Theta}_{\beta,\alpha})/2$ and quantifies how similar are layer α and β with respect to their community structure. In practice, this approach has been used to investigate the Multiplex Collaboration Network of scientists publishing in the American Physical Society journals. The main findings of this study reveal that the hierarchical organisation of physics knowledge is effectively perceived by scientists in a different way compared to the classical PACS (Physics and Astronomy Classification Scheme)[3]. Interestingly, the same

[3]It is interesting to notice that starting from 2010, The American Institute of Physics no longer maintain the PACS. Instead, they are currently working towards a new classification scheme for physics, PhySH - Physics Subject Headings [243]

indicator has been further considered when analysing the CS-Aarhus Social Network [244]. An alternative method for comparing community partitions across layers leverages on the Normalised Mutual Information (NMI) [245]. For instance, in Ref. [246] it was shown that communities belonging to different layers might be significantly different from each other (in terms of NMI) despite having similar structural features.

Nevertheless, when investigating the structure of a multilayer network, one of the most intriguing approaches was proposed in Ref. [247]. This is based on a generalisation of the map equation [200], briefly mentioned in Chapter 2.1 for single-layer systems. The main idea in this case, again, relies on the synergetic application of random walks and information-theoretic approaches. But here, a diffusion parameter is associated with inter-links, so that walkers can jump from one layer to another. As was the case with single-layer networks, functional modules can be revealed by analysing the transient state of the random walk flow dynamic. More precisely, using a remarkable encoding scheme, random walk trajectories explore the multilayer network and assign a unique code to every node in the network (based on the Huffman code). Subsequently, the string encoding the dynamics of the random walks is compressed to reveal the regularities of the multilayer networks and obtain a coarse-grained description of the system. Crucially, this method identifies overlapping modules (i.e. the case where nodes may simultaneously belong to one or many communities) that cannot be accurately recognised when analysing aggregated structures. As a proof of concept, the authors analysed both synthetic and real-world multilayer networks, including ArXiv and Pierre Auger collaboration networks (see Fig. 2.7). Notice that a possible way to determine whether certain nodes belong to the "same" community across different layers is based on examining the similarities among the partitions of the different layers using similarity indicators, such as the Jaccard [248], Rand [249], and Adjsted Rand indices [250]. More details on the topic can be found in dedicated reviews [18, 199, 204].

The most thrilling part of using information-theoretic approaches, however, regards evaluating the structural similarities between layers or, even more significantly, finding lower-dimensional

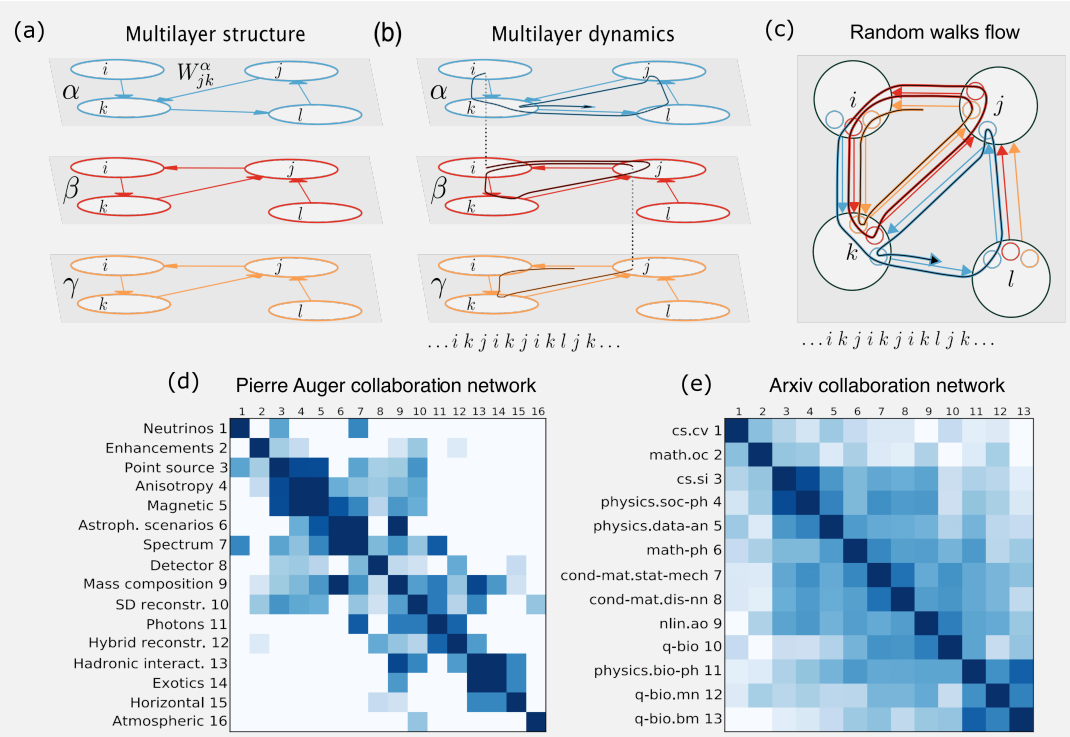

Figure 2.7. ***Overlapping modules in synthetic and real-world multilayer networks.*** *(a) A synthetic multilayer network with four nodes and three layers. (b) A random walker can either move within the same layer or randomly jump between the layers while exploring the system. (c) The random walker visits the replica nodes of the multilayer network identifying two overlapping modules with respect to flow. Indeed, walkers moving on the red and yellow links remain trapped in the upper triangle, while when crossing blue links, the walkers remain trapped in the lower triangle. In real-world multilayer networks, namely, the (d) Pierre Auger and (e) Arxiv collaboration networks, modules across layers are often in overlap with each other. The heat map reports the similarity between layers of the multiplex. This is measured as the fraction of replica nodes in different layers that are assigned to the same communities. Taken and adapted from [247].*

representations of a multiplex system. Indeed, it is in this context that most of the new information-theoretic approaches were recently proposed. As a matter of fact, how much of the information present in a multiplex network do we really need? Ultimately, our goal is to keep track of the different nature of the links only if there is a benefit in doing so. In other words, if we aim at employing economical models, relying on a multiplex formulation should be the outcome of a trade-off between the level of information and the resources required to store it. For this reason, measures aimed at quantifying the similarity between layers of a multiplex might help in this direction. In Ref. [251], for instance, two new metrics inspired by [167] and mainly

based on the Jensen-Shannon divergence were proposed to assess the node and layer diversity in a multiplex structure. With a similar argument, the network portrait divergence [169] allows comparing networks based on a graph invariants and, eventually, to compare layers of multiplex and temporal networks. Yet, whether a lower-dimensional multilayer network can exhibit the same structural and dynamic richness of the full multilayer graph is still an open question. Some concrete attempts in this direction were provided in Ref. [60], using a (quantum) information-theoretic approach, in Ref. [252–254] with approaches relying on mesoscopic similarity between layers, or alternatively in Ref. [255] with the concept of functional reducibility. In particular, the method presented in Ref. [60] leverages on a formal parallel between density operators in quantum systems and Laplacian matrices in graph theory. In this way, the concept of the von Neumann entropy of a graph is extended to the case of multiplex networks.

For our purpose, let us introduce the main elements of this method more formally. First, we consider a multiplex network $\mathcal{M} = \{A^{[1]}, \ldots, A^{[M]}\}$ with N nodes and M layers. We then introduce the (rescaled) Laplacian matrix $\mathcal{L}^{[\alpha]}$ associated with the adjacency matrix $A^{[\alpha]}$ of layer α as:

$$\mathcal{L}^{[\alpha]} = \frac{1}{\sum_{i,j} A_{ij}^{[\alpha]}} \left[D^{[\alpha]} - A^{[\alpha]} \right] \tag{2.37}$$

where $D^{[\alpha]}$ represents the diagonal matrix of the degrees of the nodes at layer α. With these elements, and with the underlying assumption that each layer is statistically independent from one another, the authors extended the definition of the von Neumann entropy for multiplex network, i.e. $H(\mathcal{M})$, as the sum of the von Neumann entropies of all its layers $h_{A^{[\alpha]}}$,i.e. :

$$H(\mathcal{M}) = \sum_{\alpha=1}^{M} h_{A^{[\alpha]}} = \sum_{\alpha=1}^{M} -Tr \left[\mathcal{L}^{[\alpha]} \log_2 \mathcal{L}^{[\alpha]} \right] = - \sum_{\alpha=1}^{M} \left[\sum_{i=1}^{N} \mu_i^{[\alpha]} \log_2 \left(\mu_i^{[\alpha]} \right) \right] \tag{2.38}$$

where $\mu_i^{[\alpha]}$ are the eigenvalues of the (rescaled) Laplacian matrix $\mathcal{L}^{[\alpha]}$ at layer α. Notice that the (rescaled) Laplacian matrix is a semi-definite positive matrix with eigenvalues summing up to 1, so that it has the same properties as density operators in quantum mechanics. In the case

of quantum systems, the Von Neumann entropy has a clear physical interpretation. Indeed, it is equal to zero when the system is in a pure state, while it is greater than zero when the system is in a mixed state. In the context of graph theory, the Von Neumann entropy of a graph has been interpreted as "the entanglement of the statistical ensemble of pure states where each pure state is one of the edges of the graph" [60]. In the case of multiplex networks, each of the layers represents one of the many possible states of the system, i.e. a network state. Notice, however, that the von Neumann entropy $H(\mathcal{M})$ actually depends on the size of the multiplex, i.e. the total number of layers. For this reason, when quantifying the von Neumann entropy of a reduced multiplex $\mathcal{C} = \left\{C^{[1]}, \ldots, C^{[X]}\right\}$ with $X \leq M$ layers, it is convenient to consider the entropy per layer of the multiplex $\mathcal{C}$ as:

$$\overline{H}(\mathcal{C}) = \frac{H(\mathcal{C})}{X} = \frac{\sum_{\alpha=1}^{X} h_{C^{[\alpha]}}}{X}. \tag{2.39}$$

With all these elements, it is now possible to quantify how a multiplex network $\mathcal{C}$ is distinguishable from its corresponding aggregate network $\mathcal{A}$ [defined in Eq. (2.8)] as:

$$q_{VN}(\mathcal{C}) = 1 - \frac{\overline{H}(\mathcal{C})}{h_{\mathcal{A}}} \tag{2.40}$$

where $h_{\mathcal{A}}$ represents the von Neumann entropy of the aggregate network $\mathcal{A}$ associated with the (possibly) reduced multiplex $\mathcal{C}$. According to Eq. (2.40), $q_{VN}(\mathcal{C}) \geq 0$ since the Von Neumann entropy of a graph is expected to increase if the graph becomes denser [60]. As a consequence, the Von Neumann entropy of the aggregate graph $h_{\mathcal{A}}$ will always be greater or equal than $\overline{H}(\mathcal{C})$. More precisely, if all the layers of a multiplex are identical, then $q_{VN} = 0$, since the aggregate and the multiplex encode the same network. By contrast, when $q_{VN} > 0$, the multiplex system is distinguishable from its aggregate. Therefore, the multiplex structure is relevant and should be preserved [60]. Yet, the only missing piece of the procedure remains: how to quantify the similarity between layers in order to aggregate the most similar and reduce the system size? To

this end, the authors defined the Jensen-Shannon divergence between layers α and β as:

$$JSD^{[\alpha,\beta]} = \sqrt{h\left(\frac{1}{2}\left[A^{[\alpha]} + A^{[\beta]}\right]\right) - \frac{1}{2}\left[h\left(A^{[\alpha]}\right) + h\left(A^{[\beta]}\right)\right]}. \tag{2.41}$$

The JSD takes values in $[0, 1]$ and satisfies all the properties of a metric if applied to qbit [60, 256]. Within this method, starting from a multiplex network $\mathcal{M}$ with M layers, the optimal configuration of aggregated layers is obtained by maximising the relative entropy $q_{VN}(\bullet)$. However, such configuration requires the enumeration of all the feasible partitions of a set of M elements (i.e. the layers), which is known to be a NP-hard problem [257]. As a consequence, the authors of Ref. [60] employed a greedy hierarchical clustering, which reduces the overall complexity to $\mathcal{O}(M^2)$. In particular, at each step of the aggregation procedure the pairs of layers yielding the smallest value of JSD are aggregated together, by taking the union of the edges of the two layers.

In short, starting from a multiplex network with M layers, at each step of the method a greedy procedure, based on the JSD between layers, allows the identification of the most similar pair of layers. All the pairwise interactions are then used to create a dendrogram containing the subsequent aggregation between layers. Finally, the optimal multiplex representation is identified by looking at the maximum value of the distinguishability function $q_{VN}(\bullet)$, i.e. the argmax$[q_{VN}(\bullet)]$ (see Fig. 2.8 for the salient steps of the method). Despite the intriguing formalism, notice that the method does not allow to reduce a multilayer representation with M layers to just one layer, even if the aggregate and the multiplex encode the same network. This is because, for the way the distinguishability function is defined, a compressed multiplex network made of only one layer will always have $q_{VN} = 0$, while all the other compressed representations with $1 \neq X \leq M$ layers will generally have $q_{VN} \geq 0$. In the extreme case of a multiplex made of identical layers, all the reduced multiplex networks $\mathcal{C}$ have a value of $q_{VN}(\mathcal{C}) = 0$ but, in this case, since the distinguishability function is identically zero, a maximum is not defined.

Nevertheless, one of the main findings of this study is that biological and social systems often show a high level of redundancy, so that these systems admit a low-dimensional representation made of only a few layers. By contrast, man-made systems such as transportation systems or technological networks purposedly avoid redundancies across layers, so that only some layers can be effectively compressed.

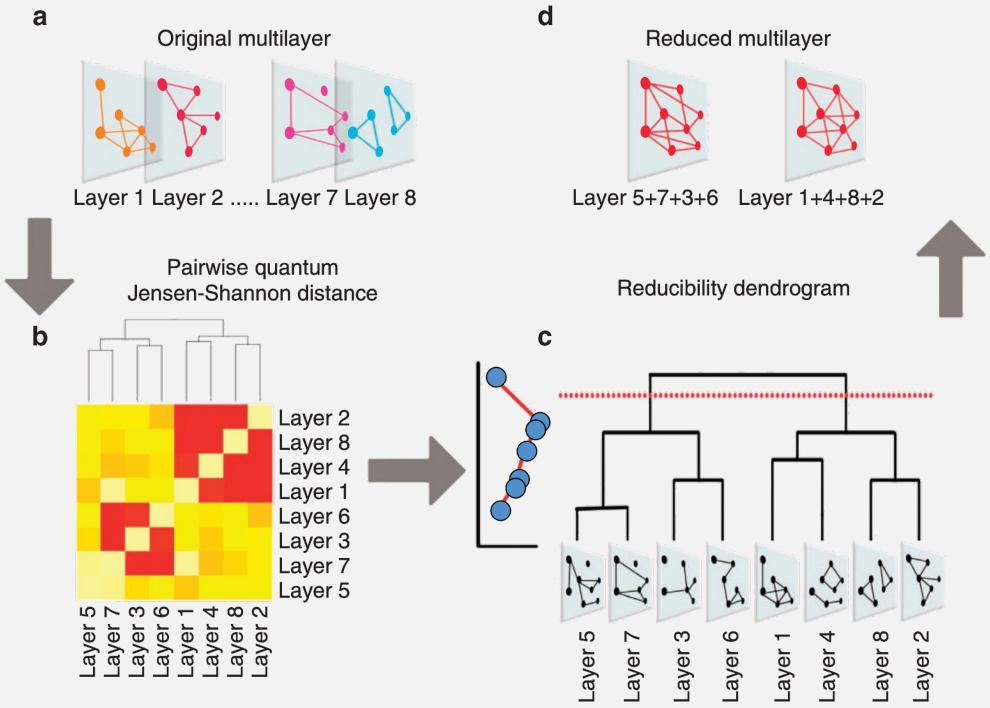

Figure 2.8. ***Diagram of the method proposed in Ref. [60] for reducing a multiplex network.*** *(a) Starting from a multiplex network $\mathcal{M}$, all the pairwise JSD between the layers are computed (b). This similarity measure allows the construction of a hierarchical dendrogram describing all the subsequent aggregation between the layers. (c) The maximum value of the distiguishability function $q_{VN}(\bullet)$ is employed as quality function to determine the optimal (reduced) multiplex (d). Reprinted from [60].*

Despite interesting results, the method proposed in Ref. [60] suffers from three shortcomings: first, it is not intended to reduce multilayer systems to low-dimensional representation made of only one layer (i.e., the aggregate network), second, it cannot be used for comparing the information content of different multilayer networks and third, there are no guarantees that the method finds low-dimensional representations that retain the same structural and dynamic behaviour of the original systems. To address the aforementioned aspects, we shall present in the

next Chapter a new method [2] to quantify the amount of information contained in a multilayer network model by using an algorithmic information theory perspective.

Chapter 3

Algorithmic complexity of multiplex networks

In this chapter, we shall present a new method, based on an algorithmic information theory perspective, to assess when the multilayer representation of a system provides a qualitatively better model than the classical single-layer aggregated network. First, we define the complexity of a multilayer network as the additional information contained in multiplex networks with respect to the corresponding single-layer graph representation. We then characterise the main properties of this complexity and show that it can be used to compare the information content of different multiplex networks. Finally, we demonstrate how our complexity measure can be used to obtain low-dimensional representations of multiplex systems, to cluster multilayer networks into a small set of meaningful super-families, and to detect tipping points in the evolution of different time-varying multilayer graphs. Notice that in the context of finding low-dimensional representations of multiplex systems, we provide extensive numerical evidence that our method outperforms the approaches described in the previous chapter. In particular, our approach overcomes some of the existing limitations of previous reducibility methods, so that it allows to:

(i) reduce a multiplex network with several layers into just a single-layer network; (ii) obtain lower-dimensional representations which retain most of the structural and dynamical properties of the original multilayer system; (iii) compare the amount of information encoded in two different multilayer systems having different number of nodes, edges, and layers; (iv) pinpoint some of the properties of the original multiplex as responsible for changes in the results of our method. As a consequence, our method represent the new state-of-the-art approach for compressing and reducing the information contained within a multiplex network. This chapter, extensively based on our work of Ref. [2], represents one of the contributions of the thesis to the field of complex networks.

3.1 Multiplex complexity

As discussed in Chapter 1.3, the concept of Kolmogorov complexity was proposed in the '60s as an algorithmic measure to quantify the complexity of a bit string. Here use the same concept to compute the algorithmic complexity of multiplex systems. In particular, we first propose a novel encoding scheme that allows us to convert a multiplex network into a bit string. Then, we quantify the complexity of a multilayer network as the ratio of the Kolmogorov complexity of the bit string associated with the multiplex and with the corresponding aggregate network.

Let us introduce in more detail all the steps of that method. As we have seen in Section 2.2, an unweighted multiplex network $\mathcal{M}$ with M layers and N nodes can be described by a set of adjacency matrices $A = \{a_{ij}^{[\alpha]}\} \in \mathbb{R}^{M \times N \times N}$. Here instead, we propose to encode the multiplex $\mathcal{M}$ in a single $N \times N$ real matrix, i.e., the *prime-weight matrix* Ω, defined as:

$$\Omega_{i,j} = \begin{cases} \prod_{\alpha \,:\, a_{ij}^{[\alpha]} = 1} p^{[\alpha]} \\ \\ 0 \qquad\qquad \text{if } a_{ij}^{[\alpha]} = 0 \;\; \forall \; \alpha = 1, \ldots, M. \end{cases} \tag{3.1}$$

The prime-weight matrix is constructed by assigning to each layer $\alpha = 1, \ldots, M$ a distinct prime number $p^{[\alpha]}$, and then defining each entry of the matrix Ω_{ij} as the product of primes associated with the layers where the edge (i, j) is present. Thus, by definition $0 \leq \Omega_{i,j} \leq \prod_{\alpha=1}^{M} p^{[\alpha]}$. Thanks to the unique factorization theorem (i.e., the product of prime numbers is unique, up to the order of the factors), the prime-weight matrix preserves the information about the nature of each tie in the multiplex. As a consequence, within this mathematical representation, the adjacency matrix of a certain layer $A^{[\alpha]}$ can be obtained from Ω by considering all the elements Ω_{ij} which are divisible by the corresponding prime $p^{[\alpha]}$. Although the choice of prime numbers does not have any impact on the definition of the prime-weight matrix, in practice it is convenient considering the smallest prime sequence $P = \{2, 3, 5, \ldots\}$. This is because the number of bits required to store the matrix Ω is $\mathcal{O}\left(N^2 M \log_2 \left[\max_\alpha \left\{p^{[\alpha]}\right\}\right]\right)$, therefore any mathematical operations, such as the Great Common Divisor (GCD) or simple divisions, might become computationally demanding when considering large primes.

Given a prime association to the M layers, i.e. $P = \{p^{[1]}, \ldots, p^{[M]}\}$, the prime-weight matrix Ω is uniquely determined. However, given a certain prime sequence P, in principle any of the $M!$ possible permutations could be considered for describing the same multiplex $\mathcal{M}$. In our case, we employ the particular prime association $\hat{P}$, which is the one that associates prime numbers to layers in increasing order of their total number of edges $K^{[\alpha]} = \frac{1}{2} \sum_{i,j} a_{ij}^{[\alpha]}$. In other words, we associate the prime number 2 to the layer with the smallest total number of edges $K^{[\alpha]}$, the prime 3 to the layer with the second-smallest total number of edges, and so on. This peculiar association, i.e. $\hat{P}$, as we will show in the next section, provides us with interesting properties when computing the complexity of multilayer networks. Unless specified, we will use $\hat{P}$ in all our numerical simulations, and we will refer to it as the *canonical prime association*. Notice that in presence of L layers having the same number of links (i.e. ties in the ranking of number of edges), the canonical prime association $\hat{P}$ is constructed in a similar fashion as detailed above, yet accounting for ties. That is, L subsequent primes of the sequence

$\hat{P} = \{p^{[1]}, \ldots, p^{[l]}, \ldots, p^{[l+L]}, \ldots, p^{[M]}\}$ are randomly assigned to the corresponding L layers and we consistently rely on the same prime association when averaging numerical simulations. In the limit case of multiplex with the same number of links on each layer, we consider only one random prime association and use that for all the numerical simulations.

With the multiplex encoding provided by Ω, we now have all the tools to map a multiplex network $\mathcal{M}$ to a bit string $s(\mathcal{M})$ [or interchangeably $s(\Omega)$]. This map is obtained by considering the bit string of the edge list associated with the prime-weight matrix Ω, where the edges are listed in lexicographic order and each edge (i, j) reports the corresponding entry Ω_{ij}. In other words, all the N distinct nodes of the multiplex are consecutively numbered from 0 to $N - 1$, and the list of tuples (i, j, Ω_{ij}) is first sorted with respect to i and then to j. Henceforth, we will refer to $s(\Omega)$ as the bit string representation of the matrix Ω. Note, however, that such a bit string representation $s(T)$ can be used in principle for any matrix T. In particular, for any bit string s we can rely on the Kolmogorov complexity $KC(s)$ as a measure of the information content contained in s. Again, because $KC(s)$ cannot be computed, it is usually approximated by means of a compression algorithm. With these elements, we define the complexity $\mathcal{C}(\mathcal{M})$ of a multilayer network $\mathcal{M}$ as the ratio between the Kolmogorov complexity of $\mathcal{M}$ and the Kolmogorov complexity of the single-layer aggregate graph W, i.e.:

$$\mathcal{C}(\mathcal{M}) = \frac{KC(\Omega)}{KC(W)} = \frac{KC(s(\Omega))}{KC(s(W))}. \tag{3.2}$$

where W is obtained by considering the aggregate binary matrix multiplied by the largest entry of Ω:

$$W_{ij} = \begin{cases} \max\limits_{i,j}\{\Omega_{ij}\} & \text{if } \Omega_{ij} \neq 0 \\ 0 & \text{otherwise.} \end{cases} \tag{3.3}$$

In other terms, $KC(\Omega)$ represents the Kolmogorov complexity of the bit string representation $s(\Omega)$ of the prime-weight matrix Ω, while $KC(W)$ is the Kolmogorov complexity of the bit string

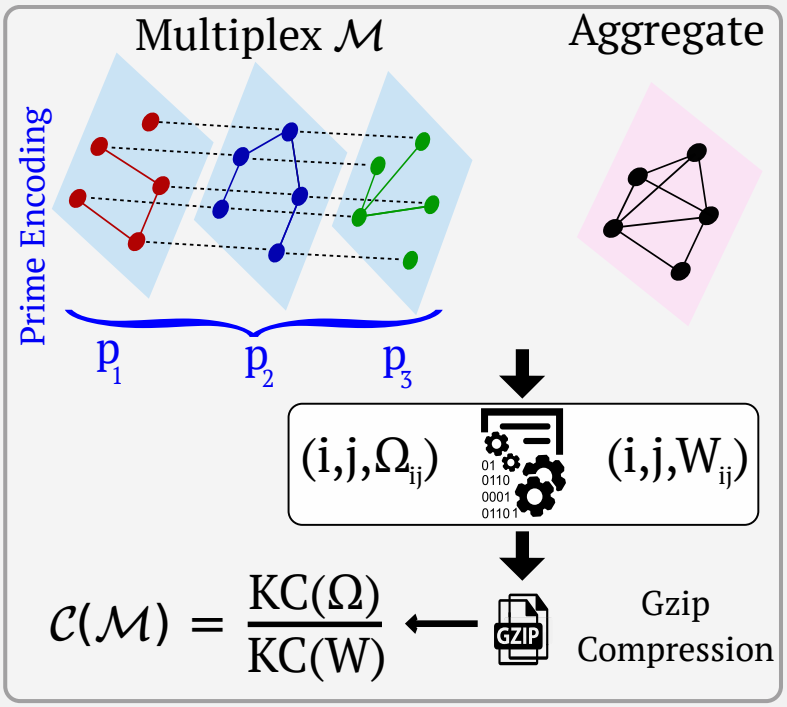

Figure 3.1. ***Schematic illustration of the definition of multiplex complexity*** $\mathcal{C}(\mathcal{M})$***.*** *We define the complexity of a multiplex network $\mathcal{C}(\mathcal{M})$ as the ratio between the Kolmogorov complexity of the multiplex and the Kolmogorov complexity of the associated single-layer aggregated graph W. The multiplex $\mathcal{M}$ is mapped into a string of bits by means of the prime-weight matrix Ω. Given the non-computability of the Kolmogorov complexity, we compute the size of the compressed string of bits associated with each object to obtain an upper-bound of the real algorithmic complexity value of the string. A common way to obtain an upper bound is by computing the length of the string compressed through the gzip algorithm.*

representation $s(W)$ of the aggregate graph W associated with $\mathcal{M}$[1]. In our case, we consider the size of the compressed bit string representation of the two matrices (obtained by means of the gzip algorithm [258]) as an approximation of the true values of Kolmogorov complexity (see Fig. 3.1 for a diagram showing all the steps).

The measure of complexity $\mathcal{C}(\mathcal{M})$ effectively estimates the additional amount of information encoded in the multiplex $\mathcal{M}$ with respect to the amount required to describe the corresponding aggregate network. As a particular case, we obtain a value of $\mathcal{C}(\mathcal{M}) = 1$ when the multiplex $\mathcal{M}$ consists of identical layers. In this setting, in fact, the multiplex representation does not

[1] In the next sections, we shall see that other possible definitions of the aggregate matrix W do not alter the qualitative behaviour of the complexity function $\mathcal{C}(\mathcal{M})$

provide any new information with respect to the single-layer graph. In the general case, the complexity $\mathcal{C}(\mathcal{M}) \geq 1$ since the number of possible configurations of edges across all the M layers of a multiplex usually requires the encoding of more than just one symbol. More precisely, $\mathcal{C}(\mathcal{M}) \approx 1$ generally implies that the single-layer representation and the multiplex one are very similar, or equivalently, that the information encoded in the multiplex is not adding much to the one present in the aggregate graph. By contrast, $\mathcal{C}(\mathcal{M}) > 1$ describes a proper multilayer model, so that the eventual aggregations of layers into a single-layer graph might possibly discard some important information. Clearly, the larger the value of $\mathcal{C}(\mathcal{M})$, the greater the amount of information contained in the multiplex model.

The essential factors contributing to the complexity measure are related to the placement of edges and their pattern of connection across the layers of the multiplex. In fact, these two factors contribute to the number of distinct symbols actually present in the prime-weight matrix Ω. Unsurprisingly, the greater the number of distinct symbols and of connected pairs of nodes in a multiplex, the larger the value of the complexity $\mathcal{C}(\mathcal{M})$. The number of symbols in Ω corresponds to the number of different multiplex motifs with two nodes [259], or alternatively to the number of different multilinks present in the multiplex [211, 230]. Notice that by using a binary representation, the amount of information needed to describe the multiplex network is actually lower than by using prime numbers. For instance, if a link appears in three layers, the amount of information required to encode such information using prime numbers is $2 \times 3 \times 5 = 30$, which requires 5 bits. By contrast, using the binary representation, the same information can be encoded as $1 + 2 + 4 = 7$ in just 3 bits. However, the prime number representation provides a number of interesting features that are not straightforward to obtain using the binary representation. That is, (i) links with integer weights w_{ij} at layer α can be easily represented in a unique way as exponent of primes (i.e. the links can be represented as $\omega_{ij} = (p^{[\alpha]})^{w_{ij}}$); (ii) in a logarithmic space, the prime factorisation of a number can be seen as a "linear combination" and, as a result, each link of a multiplex having M layers can be represented as a point in a M-

dimensional space generated by the base $\{\log p^{[1]}, \log p^{[2]}, \ldots, \log p^{[M]}\}$. For the reasons detailed above, we have chosen to rely on the encoding using prime numbers rather than the binary representation.

3.2 Properties of multiplex complexity

3.2.1 Prime association

To support the choice of the canonical prime association $\hat{P}$, we report in Fig. 3.2 the distribution of complexity obtained when reshuffling the prime association of four different multiplex networks with $M = 10$ ER layers having $N = 1000$ nodes. While in the case of a multiplex network $\bar{\mathcal{M}}$ with layers having an increasing number of links $K^{[\alpha]} = 1000 + 200 * \alpha$, with $\alpha = 0, \ldots, 9$ (panel 3.2a), the canonical prime association is located in the rightmost tail of the distribution (z-score $Z_{\mathcal{C}_{\hat{P}}} = 2.1349$). In a slightly different multiplex – same as $\bar{\mathcal{M}}$, yet obtained by replacing layers

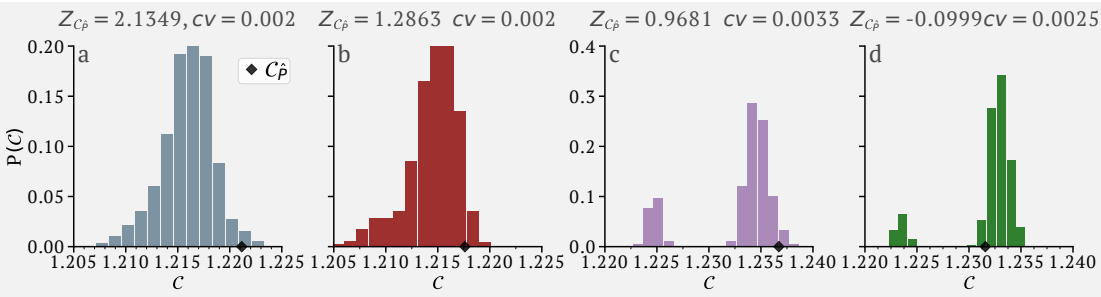

Figure 3.2. ***Impact of prime relabelling on the multiplex complexity*** $\mathcal{C}$**.** *Distribution of complexity* $\mathcal{C}$ *over* 10^5 *random prime associations for four synthetic multiplex networks with* $M = 10$ *independent ER layers having* $N = 1000$ *nodes and different number of links. Namely, a multiplex having* $K^{[\alpha]} = 1000 + 200 * \alpha$ *links, with* $\alpha = 0, \ldots, 9$ *(panel a), a multiplex as in panel (a) but with five layers having the same number of links (panel b), a multiplex with comparable amount of links between layers (each layer is sampled with the ER model with* $p = 0.005$ *- panel c), a multiplex with comparable amount of links and five layers having the same number of links (panel d). Remarkably, the canonical prime association (black diamond) is usually located above the average value of the multiplex complexity (z-score greater than 0.5), yet in the case of a multiplex system with comparable number of links between layers and in presence of layers with the same number of links, the value of complexity for the canonical prime association is lower than the average (z-score is negative).*

Multiplex	$\mathcal{C}_{\hat{P}}$	μ	σ	$ste(\mu)$	cv	$\frac{\sigma}{\mu\cdot\sqrt{\langle N\rangle}}$	$Z_{\mathcal{C}_{\hat{P}}}$
London Tube	1.128292	1.115452	0.007446	0.000074	0.006675	0.000067	1.724483
Barcelona Tube	1.126812	1.119959	0.007222	0.000072	0.006448	0.000064	0.948881
Beijing Tube	1.120737	1.122686	0.010934	0.000109	0.009739	0.000097	-0.178297
Berlin Tube	1.107717	1.098907	0.009024	0.000090	0.008212	0.000082	0.976258
Airports North America	1.263749	1.250040	0.014373	0.000144	0.011498	0.000115	0.953842
Airports Europe	1.416298	1.352806	0.020549	0.000205	0.015190	0.000152	3.089739
Airports Asia	1.634686	1.591643	0.030954	0.000310	0.019448	0.000194	1.390522
Airports South America	1.306878	1.266330	0.019676	0.000197	0.015537	0.000155	2.060841
Airports Oceania	1.191390	1.139328	0.015222	0.000152	0.013361	0.000134	3.420186
Airports Africa	1.264403	1.230097	0.016762	0.000168	0.013626	0.000136	2.046697
EU airlines	1.218266	1.200618	0.016453	0.000165	0.013704	0.000137	1.072666
Train UK	1.019208	1.029434	0.017487	0.000175	0.016987	0.000170	-0.584796
APS countries	1.625382	1.590660	0.015224	0.000152	0.009571	0.000096	2.280844
Aarhus network	1.294118	1.284601	0.008598	0.000086	0.006693	0.000067	1.106912
Terrorist network	1.156780	1.145212	0.006786	0.000068	0.005925	0.000059	1.704832
Pierre Auger collab.	0.989371	0.997252	0.008041	0.000080	0.008063	0.000081	-0.980117
Arabidopsis	1.015857	1.017097	0.005778	0.000058	0.005681	0.000057	-0.214578
Candida	1.029647	1.033560	0.008517	0.000085	0.008241	0.000082	-0.459470
Celegans	1.020054	0.999806	0.010128	0.000101	0.010130	0.000101	1.999236
Drosophila	1.008709	1.021645	0.011899	0.000119	0.011647	0.000116	-1.087131
Gallus	1.049566	1.054286	0.012510	0.000125	0.011866	0.000119	-0.377249
Human Herpes-4	1.066253	1.065862	0.004286	0.000043	0.004021	0.000040	0.091231
Human HIV-1	1.033686	1.029318	0.016341	0.000163	0.015876	0.000159	0.267315
Mus	1.039883	1.039828	0.006973	0.000070	0.006706	0.000067	0.007934
Oryctolagus	1.009524	1.014351	0.007962	0.000080	0.007850	0.000078	-0.606175
Plasmodium	0.988591	0.991987	0.006902	0.000069	0.006957	0.000070	-0.492102
Rattus	1.048545	1.033855	0.011312	0.000113	0.010942	0.000109	1.298544
S. Pombe	1.061032	1.063236	0.007830	0.000078	0.007364	0.000074	-0.281524
Xenopus	1.053121	1.060337	0.006209	0.000062	0.005856	0.000059	-1.162198

Table 3.1. ***Statistics of multiplex complexity*** $\mathcal{C}(\mathcal{M})$ ***for different real-world multiplex over 10000 prime-layer associations.*** *From left to right, we report the complexity obtained using the canonical prime association* $\mathcal{C}_{\hat{P}}$*, the mean of the distribution when re-shuffling the prime association* μ*, the standard deviation* σ*, the standard error of the mean* $\mathrm{ste}(\mu)$*, the coefficient of variation* $cv = \frac{\sigma}{\mu}$*, and the coefficient of variation over the sample* $\frac{\sigma}{\mu\cdot\sqrt{\langle N\rangle}}$*. The last column reports the z-score of the canonical prime association* $\mathcal{C}_{\hat{P}}$*. The real-world multiplex networks are the same used in Ref. [60].*

with index $\alpha = 4, \ldots, 8$ with ER graphs having $K^{[\alpha]} = 2000$ links (i.e. the total number of links in the system is the same as in $\bar{\mathcal{M}}$) – the canonical prime association is only located in the right tail of the distribution with a z-score equal to $Z_{\mathcal{C}_{\hat{P}}} = 1.2863$ (panel 3.2b). However, when the number of links in the multiplex are comparable (panel 3.2c), and some of the layers even have identical number of links (panel 3.2d), the z-score of the canonical prime association assumes far lower values. Yet, the distributions of complexity of the four synthetic multiplex networks analysed have a coefficient of variation $cv = \frac{\sigma}{\mu}$ which is smaller than 0.5%, indicating that the distributions have a very low variance. In addition, we also report in Table 3.1 some statistics of the distribution of complexity $\mathcal{C}(\mathcal{M})$ over 10000 prime associations for all the real-

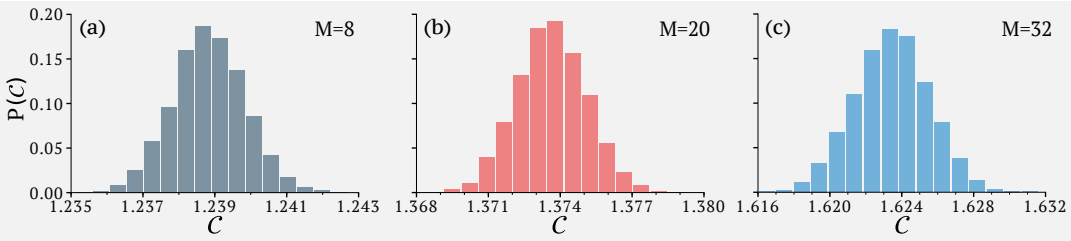

Figure 3.3. ***Impact of node relabelling on the multiplex complexity*** $\mathcal{C}$. *Distribution of complexity* $\mathcal{C}$ *over 10000 random independent node labellings for three synthetic multiplex networks with* $N = 10000$, $\langle k \rangle = 6$, $o \approx 0.6$, *and different number of layers [(a):* $M = 8$, *(b):* $M = 20$, *(c):* $M = 32$*]. The values of* $\mathcal{C}(\mathcal{M})$ *are normally distributed, as confirmed by the Anderson-Darling and Cramer Von Mises tests (resp.* $p \approx 0.79$ *and* $p \approx 0.71$ *for* $M = 8$, $p \approx 0.97$ *and* $p \approx 0.93$ *for* $M = 20$, $p \approx 0.75$ *and* $p \approx 0.76$ *for* $M = 32$*). In addition, it is worth mentioning that the coefficient of variation for all these distributions is approximately* 0.2%, *indicating a very low dispersion around the mean.*

world multiplex analysed in Ref. [60]. As expected, the prime-weight matrix encoding depends on the chosen assignment of primes to layers. Nevertheless, it is worth noting that the distribution of complexity for all the multiplex considered is quite peaked around the mean (very small standard deviation), and has a coefficient of variation $cv = \frac{\sigma}{\mu}$ which is smaller than 1% for almost all the distributions. In particular, in presence of multiplex with distinct number of links, the value of complexity obtained using the prime canonical association $\mathcal{C}_{\hat{P}}$ is usually greater than the average value of the corresponding distribution, with some exceptions given by multiplex networks having several layers with the same number of links.

3.2.2 Node relabelling

Here we study the effect of node relabelling on the value of complexity $\mathcal{C}(\mathcal{M})$, by computing the distribution of complexity over 10000 independent random node labellings for both synthetic and real-world multiplex networks. In Figure 3.3 we report the distribution of complexity for three synthetic multiplex networks having the same number of nodes and different number of layers (resp. $M = 8, 20, 32$). In all the three cases, we find that the distribution of complexity is normally distributed (confirmed by the Anderson-Darling and Cramer Von Mises tests [260]) and has a coefficient of variation (cv) less than 0.2%, indicating that the distribution is indeed

Multiplex	μ	σ	$ste(\mu)$	cv	$\frac{\sigma}{\mu \cdot \sqrt{(N)}}$
Synthetic 8 layers	1.238824	0.001007	0.000010	0.000813	0.000008
Synthetic 20 layers	1.373632	0.001465	0.000015	0.001067	0.000011
Synthetic 32 layers	1.623441	0.002087	0.000021	0.001285	0.000013
London Tube	1.125283	0.009110	0.000091	0.008096	0.000081
Barcelona Tube	1.152427	0.013088	0.000131	0.011357	0.000114
Beijing Tube	1.140638	0.010211	0.000102	0.008952	0.000090
Berlin Tube	1.111365	0.011885	0.000119	0.010694	0.000107
Airports North America	1.270578	0.004499	0.000045	0.003541	0.000035
Airports Europe	1.412997	0.005673	0.000057	0.004015	0.000040
Airports Asia	1.638689	0.005555	0.000056	0.003390	0.000034
Airports South America	1.324929	0.009317	0.000093	0.007032	0.000070
Airports Oceania	1.191850	0.010755	0.000108	0.009024	0.000090
Airports Africa	1.273712	0.010209	0.000102	0.008015	0.000080
EU airlines	1.232830	0.006486	0.000065	0.005261	0.000053
Train UK	1.018875	0.003015	0.000030	0.002959	0.000030
APS countries	1.617794	0.012755	0.000128	0.007884	0.000079
Aarhus network	1.291076	0.018428	0.000184	0.014274	0.000143
Terrorist network	1.165705	0.013456	0.000135	0.011543	0.000115
Pierre Auger collab.	1.018169	0.005196	0.000052	0.005103	0.000051
Arabidopsis	1.023403	0.001587	0.000016	0.001551	0.000016
Candida	1.030212	0.010781	0.000108	0.010465	0.000105
Celegans	1.022841	0.002089	0.000021	0.002043	0.000020
Drosophila	1.010760	0.000851	0.000009	0.000842	0.000008
Gallus	1.042861	0.008910	0.000089	0.008544	0.000085
Human Herpes-4	1.063347	0.014228	0.000142	0.013381	0.000134
Human HIV-1	1.072567	0.007249	0.000072	0.006759	0.000068
Mus	1.041120	0.001317	0.000013	0.001265	0.000013
Oryctolagus	1.025862	0.011836	0.000118	0.011538	0.000115
Plasmodium	0.986594	0.002003	0.000020	0.002030	0.000020
Rattus	1.040003	0.002909	0.000029	0.002797	0.000028
S. Cerevisiae	1.122325	0.000883	0.000009	0.000786	0.000008
S. Pombe	1.067023	0.001699	0.000017	0.001592	0.000016
Xenopus	1.071017	0.007500	0.000075	0.007002	0.000070

Table 3.2. ***Statistics of multiplex complexity*** $\mathcal{C}(\mathcal{M})$ ***for several synthetic and real-world multiplex over 10000 random independent node labellings.*** *From left to right, we report the mean of the distribution* μ*, the standard deviation* σ*, the standard error for the mean* $\mathrm{ste}(\mu)$*, the coefficient of variation* cv*, and the coefficient of variation over the sample* $\frac{\sigma}{\mu \cdot \sqrt{(N)}}$*.*

quite peaked around its mean.

The complexity distribution for real-world multiplex networks behaves in a similar way. In Table 3.2 we report some statistics for the distribution of complexity over node relabellings for the real-world multiplex networks presented in Ref. [60]. Also in this case, the distribution of complexity is peaked around the mean with a small variance for almost all the cases. To account for the intrinsic stochasticity due to node relabelling, in all the simulations presented in this chapter, we always consider the mean Complexity, which is obtained by averaging the value of

$\mathcal{C}(\mathcal{M})$ over 10^3 independent node labellings.

3.3 Behaviour of the multiplex complexity

As seen in Section 2.3, one possible way to keep track of the redundancies in a multiplex network is by means of the overlapping matrix $\mathcal{O} = \{o_{ij}\}$. Here, we condense the information related to the amount of overlap in a multiplex into a single measure, which accounts for the expected number of layers in which a pair of nodes is connected by an edge [2, 43]:

$$o_s = \frac{\sum_{i,j}^{N} o_{ij}}{M \sum_{i,j}^{N} \theta(o_{ij})} \tag{3.4}$$

where $\theta(x)$ is the Heaviside function, i.e. $\theta(x) = 1$ if $x > 0$, and $1/M \leq o_s \leq 1$, while the quantity $o_{ij} = \sum_{\alpha} a_{ij}^{[\alpha]}$ represents the number of times the edge (i, j) is present in the multiplex. Notice that $o_s = 1/M$ when each edge (i, j) appears exactly once in the multiplex, while $o_s = 1$ when all the layers are identical. To properly compare multiplex networks with a different number of layers, we also introduce the linear transformation $f(o_s) := \frac{M}{M-1}\left(o_s - \frac{1}{M}\right)$ to map o_s into the interval $[0, 1]$, and thus define the *structural overlap* o of a multiplex as:

$$o = \frac{M}{M-1}\left(o_s - \frac{1}{M}\right). \tag{3.5}$$

We first examined the effect that the structural overlap has on the complexity function $\mathcal{C}(\mathcal{M})$. We start by considering an ensemble of synthetic multiplex networks consisting of different number of layers, where the total number of nodes and the average node degree on each layer are kept fixed, while the structural overlap is tunable. In practice, to tune the overlap of a multiplex network we generalise the rewiring procedure introduced in Ref. [41]. We start by considering a multiplex network with M identical layers (i.e. multiplex with structural overlap

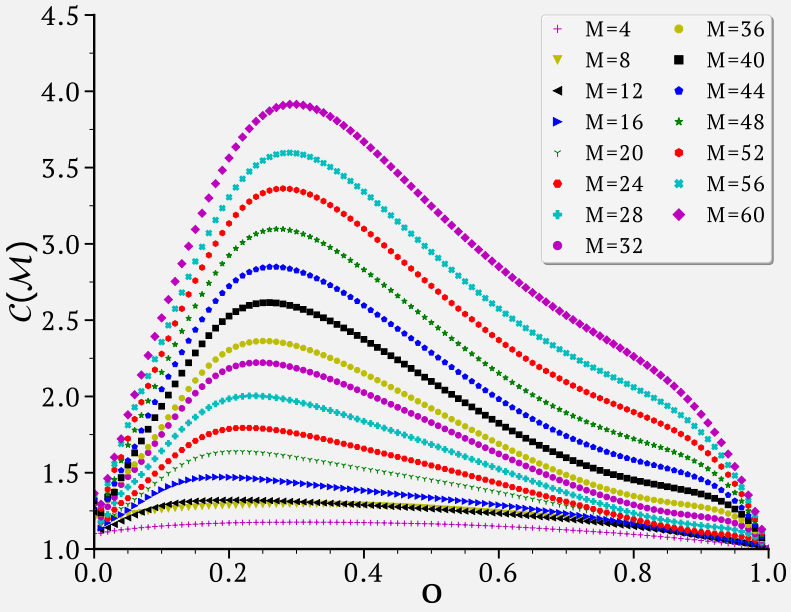

Figure 3.4. ***Behaviour of the multiplex complexity as a function of the structural overlap*** *o. We report $\mathcal{C}(\mathcal{M})$ as a function of the structural overlap o for an ensemble of multiplex networks consisting of ER random graphs on each layer with $N = 10000$ nodes, $\langle k \rangle = 6$, and variable number of layers. Remarkably, we observe a non-monotonic behaviour of $\mathcal{C}(\mathcal{M})$ as a function of the structural overlap. As a consequence, there exists an optimal amount of information that a multiplex representation can encode with respect to its corresponding aggregate graph (i.e. this is obtained in the range $o \in [0.15, 0.4]$).*

$o = 1$). Then, we iteratively reduce the structural overlap to $o = 0$ by uniformly rewiring the edges on each layer. The rewiring procedure is performed by uniformly selecting a layer α among the M possibilities, then selecting a pair of edges and swapping their end-points uniformly at random. As a result, the rewiring procedure preserves the degree sequence at each layer. We also develop an algorithm to increase the structural overlap in a multiplex network in a similar fashion of the procedure described for reducing the overlap. In this case, however, the rewiring of an edge is always rejected if results in a decrease of the edge overlap of at least one.

Fig. 3.4 illustrates the complexity function $\mathcal{C}(\mathcal{M})$ as a function of the structural overlap o, when considering an ensemble of multiplex network consisting of ER random graphs on each layer with $N = 10000$ nodes and $\langle k \rangle = 6$.

As previously mentioned, when the multiplex network has M identical layers (i.e., $o = 1$),

then $\mathcal{C}(\mathcal{M}) = 1$. Yet, when we start rewiring the edges in order to decrease the overlap of the multiplex with identical layers, we expect an increase in complexity. By contrast, when $o \approx 0$ the number of distinct links in the multiplex representation is approximately equal to the number of edges in the aggregate matrix. We thus expect to see a complexity value $\mathcal{C}(\mathcal{M})$ close to one, since the values of Kolmogorov complexity of both the numerator and denominator of Eq. (3.2) are approximately the same. This is what we observe in Fig. 3.4 for the two extremal values of overlap, i.e. $o \approx 0$ and $o \approx 1$. The most interesting feature comes when we examine intermediate values of overlap. In the interval $o \in [0.15, 0.4]$ there exists an optimal amount of information that a certain multiplex representation can encode with respect to its corresponding aggregate graph. Thus, according to our complexity measure, for these values of overlap a multiplex representation is always preferable compared to its aggregate network. Remarkably, $\mathcal{C}(\mathcal{M})$ shares the same salient traits of a "good" complexity function, as presented in Fig. 1.2 of Chapter 1.2. That is, the complexity is low when the multiplex is either highly ordered ($o \approx 1$) or disordered ($o \approx 0$), while it takes high values for multiplex with intermediate values of structural overlap.

This peculiar characteristic comes from the way the complexity $\mathcal{C}(\mathcal{M})$ is defined, i.e. as the ratio between two Kolmogorov complexities. In a way, such definition resembles the physical complexity mentioned in Chapter 1.2, that is, the complexity of a string cannot be considered in isolation, but it must be compared within its "environment". By only relying on either the Kolmogorov complexity of the numerator or denominator of Eq. 3.2, in fact, we would not be able to assign a low value of complexity for highly disordered multiplex networks ($o \approx 0$).

3.3.1 Effect of network rewiring

To fully characterise the behaviour of the complexity function $\mathcal{C}(\mathcal{M})$, we also consider a different ensemble of networks where we iteratively increase and decrease the structural overlap using the rewiring procedure described above. When increasing (resp. decreasing) the overlap, we

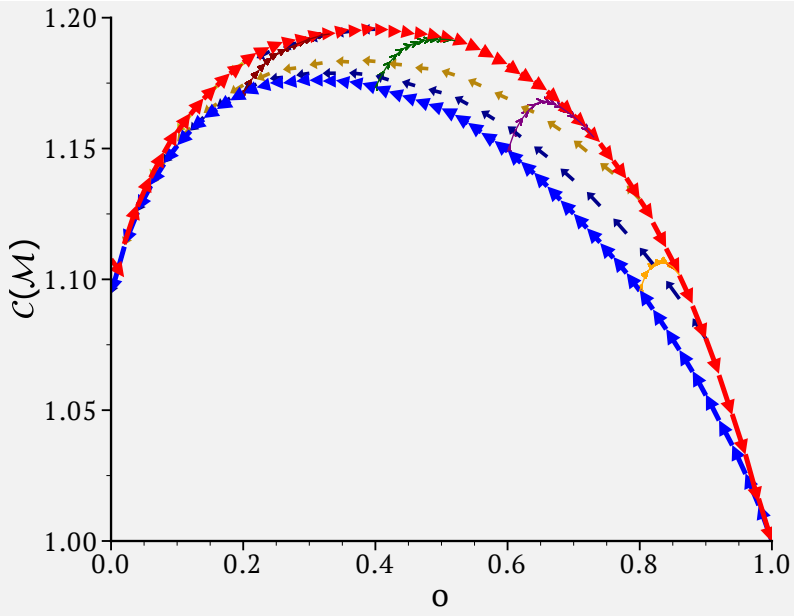

Figure 3.5. ***Hysteresis loop for the multiplex complexity*** $\mathcal{C}(\mathcal{M})$ ***as a function of the structural overlap*** o. *A hysteresis loop of* $\mathcal{C}(\mathcal{M})$ *appear for a multiplex with* $N = 10000$ *nodes, and* $M = 4$ *layers with average degree* $\langle k \rangle = 6$. *We report with red arrows the trajectory observed when the structural edge overlap is decreased, while with the blue arrows the trajectories observed when the structural overlap is increased.*

observe two distinct trajectories in the $[o - \mathcal{C}(\mathcal{M})]$ plane, which characterise a hysteresis loop (see Fig. 3.5). In particular, we start by considering a multiplex network consisting of four identical ER random graph with $N = 10000$ nodes and average degree $\langle k \rangle = 6$. We then compute the complexity measure when reducing the overlap of the multiplex by rewiring its edges until we obtain a multiplex with $o = 0$. Then, we start the rewiring procedure again to increase the structural overlap until the system results in a multiplex network with $o = 1$. The two distinct trajectories indicate that the procedures used for decreasing (resp. increasing) the edge overlap are not ergodic. This is related to the profound difference between the ways in which the overlap is created and destroyed. In fact, the number of ways the overlap can be reduced at random starting from a multiplex network with identical layers is far more numerous than the number of ways the overlap is increased at random starting from a multiplex network with all distinct layers (i.e., structural overlap equal to zero). Thus, extreme care is required for the rewiring process of a multiplex network. As we will see later in the thesis, knowing this small caveat

comes in handy when dealing with null-models that aim at maintaining the structural overlap of empirical/synthetic multiplex networks.

3.3.2 Effect of aggregate network and compression algorithm

In the computation of $\mathcal{C}(\mathcal{M})$, we fixed two particular features: (i) we rely on the gzip algorithm [258] for approximating the Kolmogorov complexity of the edge lists $s(\Omega)$ and $s(W)$, and (ii) we choose a particular form of the aggregate matrix W [defined in Eq. (3.3)] to ensure $\mathcal{C}(\mathcal{M}) = 1$ when all the layers of a multiplex are identical. But these choices are not the only possibilities. Other formulations of the aggregate matrix W could be used, in principle. Similarly, any compression algorithm can be used instead of gzip for computing an upper bound of the Kolmogorov complexity.

To investigate the dependence of the complexity function on these two aspects, we consider two alternative common compression algorithms, namely, bzip2 [261] and lz4 [262], and two alternative aggregate representations that could be actually used in place of our original definition of W:

$$W'_{ij} = o_{ij} = \sum_{\alpha=1}^{M} a_{ij}^{[\alpha]} \tag{3.6}$$

or:

$$W''_{ij} = \begin{cases} 2^{o_{ij}} & \text{if } o_{ij} > 0 \\ 0 & \text{otherwise.} \end{cases} \tag{3.7}$$

We report the comparisons between the two alternative formulations/algorithms in Fig. 3.6. In both panels, we observe that although there are some minor quantitative differences, the qualitative behaviour of the complexity function remains similar. In other words, we still obtain a non-monotonic behaviour of $\mathcal{C}(\mathcal{M})$ as a function of the structural overlap o.

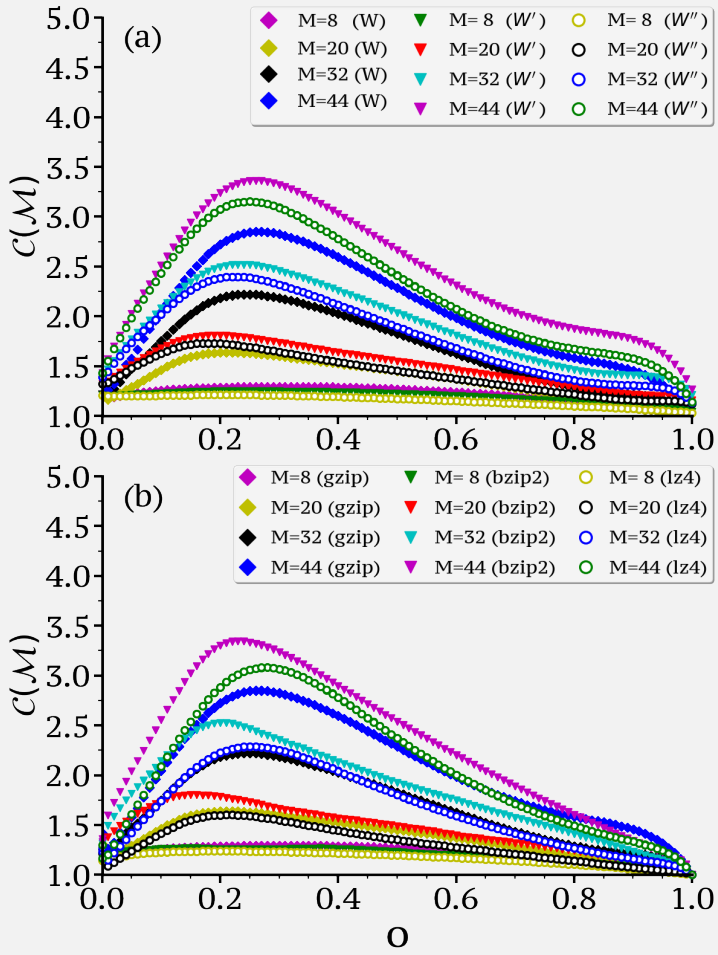

Figure 3.6. ***Impact of the weighted aggregate representations and compression protocols on the complexity*** $\mathcal{C}(\mathcal{M})$. *We report, for an ensemble of synthetic multiplex networks with a variable number of layers and tunable structural overlap* o *(*$N = 10000$ *nodes and* $\langle k \rangle = 6$*), the complexity* $\mathcal{C}(\mathcal{M})$ *as a function of the structural overlap. (a) For all the three weighted aggregate representations,* W W'*, and* W''*, we observe a similar behaviour when computing the Kolmogorov Complexity of binary strings. Despite minor quantitative differences, also in this case the complexity* $\mathcal{C}(\mathcal{M})$ *has a non-monotonic behaviour as a function of the structural overlap, for any value of* M*. (b) Similar considerations hold when considering different compression protocols.*

3.3.3 Effect of network topology and average degree

In Figure 3.7(a) we report the value of complexity in the ensembles of synthetic multiplex networks with different average degree on each layer, where the total number of nodes and the number of layers are kept fixed (N = 10000, M = 20), while the the structural overlap o is tunable. As expected, the complexity increases slightly as a function of the average degree $\langle k \rangle$,

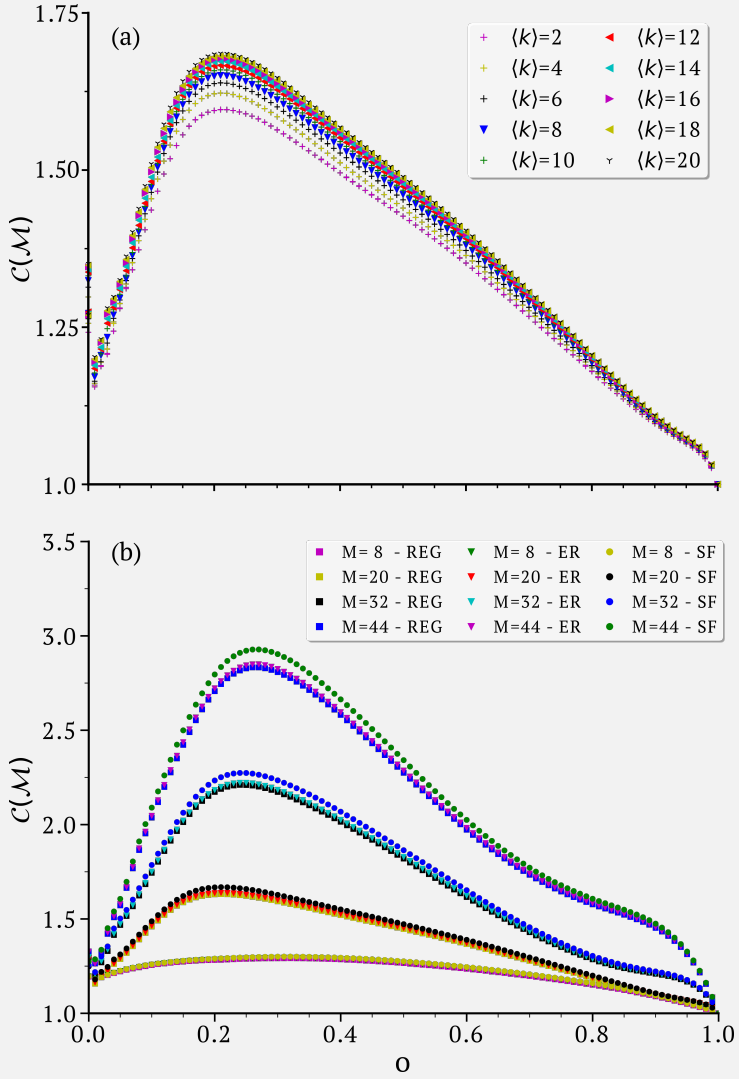

Figure 3.7. ***Impact of the average degree and network topology on the complexity*** $\mathcal{C}(\mathcal{M})$. *(a) We report the multiplex complexity* $\mathcal{C}(\mathcal{M})$ *for an ensembles of synthetic multiplex networks with* $M = 20$ *layers, variable average degree* $\langle k \rangle$ *and tunable structural overlap* o*. Each layer is an Erdös-Rényi graph with* $N = 10000$ *nodes. The* $\mathcal{C}(\mathcal{M})$ *proportionally increases as a function of* $\langle k \rangle$ *while maintaining the same optimal value of complexity around* $o \approx 0.21$*. (b) For an ensembles of synthetic multiplex networks with* $\langle k \rangle = 6$ *and tunable structural overlap* o*, we report the values complexity for different topologies, namely, regular networks, ER random graph, and scale free networks with* $\gamma = 2.7$*. As one would expect, it appears that scale-free multiplex networks have a higher complexity compared to multiplex having ER or regular topologies.*

while the shape of the curve and the position of the maximum remain unchanged.

We further investigate the impact of different network topologies on the complexity. We report in Figure 3.7(b) the comparison of complexity in the ensembles of synthetic multiplex

networks with different topologies, namely, regular, Erdös-Rényi , and scale-free networks ($\gamma = 2.7$) with the same average degree. In this case, we observe a slight difference in the values of complexity depending on the topology. That is, scale-free multiplex networks have a slightly higher complexity compared to multiplex systems, whose layers are ER or regular topologies.

3.4 Algorithmic reducibility of multiplex networks

One of the most intriguing aspects of adopting a multiplex representation is that it usually conveys more information than the corresponding aggregate graph. Yet, a considerable chunk of multi-dimensional data sets is normally redundant and can sometimes be neglected. If we consider a straightforward mapping of each type of relation of the multi-dimensional data set into a layer of a multiplex framework, this might eventually result in a redundant representation of the original system. This is often inconvenient, since computing even simple multilayer structural descriptors, such as clustering coefficient, average shortest path or any centrality measure based on paths, scale super-linearly or exponentially as a function of the number of layers [216]. Hence, it is always desirable to deal with parsimonious models and examine them under Occam's razor. Here, we leverage the complexity measure $\mathcal{C}(\mathcal{M})$ as a powerful tool to tackle the reducibility problem, i.e. to determine the optimal amount of layers in a multiplex representation which preserves as much structural and dynamical information as possible about the original system. In particular, if we consider a multiplex network $\mathcal{M}$ with M layers and we aggregate some of the layers, we will obtain a reduced representation $\mathcal{X}$ of the original system with $X \leq M$ layers. Thus, we propose to quantify the normalised information content in the multiplex as:

$$q(\mathcal{X}) = \frac{\mathcal{C}(\mathcal{X})}{\log K_{\mathcal{X}}} \tag{3.8}$$

where $K_{\mathcal{X}}$ is the total number of links in the multiplex $\mathcal{X}$. At first glance, the normalisation factor seems superfluous. But, $\log K_{\mathcal{X}}$ allows us to properly account for the Kolmogorov complexity

of bit strings of different length. In particular, the algorithmic complexity of an ensemble of bit strings of length n grows as $\log n + \mathcal{O}(1)$ [85]. In our setting, the length of the bit strings is proportional, on average, to the number of links in the multiplex. As a result, a system with a larger number of links is expected to have a larger value of multiplex complexity. This is a technical inconvenience when dealing with the reducibility problem. Clearly, a multiplex with a larger number of layers will usually have a larger amount of edges and will be encoded in a longer string. For this reason, dividing the value of multiplex complexity by $\log(K_{\mathcal{X}})$ allows us to safely compare low-dimensional representations of a multiplex network with a different number of layers (and links).

The function $q(\mathcal{X})$ can be regarded as a quality function, since it quantifies the information content that a certain multiplex network $\mathcal{X}$ encodes with respect to its weighted aggregate representation $W_{\mathcal{X}}$, as defined in Eq. (3.3). Thus, it can be effectively employed during a reducibility procedure for selecting the optimal compressed representation. In other words, we aim to select the (possibly reduced) multiplex representation that yields the highest value of $q(\mathcal{X})$, or equivalently, to find $\operatorname{argmax}[q(\bullet)]$. Notice that, a system consisting of all identical layers will always have its maximum value of quality function $q(\mathcal{X})$ at $1 = X \leq M$ layers, since the multiplex and the aggregate network representations are equivalent and have the same Kolmogorov complexity.

Computing the global maximum of the quality function $q(\bullet)$ is in general computationally unfeasible since it requires enumerating all the possible partitions of M layers. This is an NP-hard problem that requires a number of operations that scales super-exponentially with M (also known as the Bell number [257]). For this reason, a conventional way to overcome this issue is to rely on a greedy aggregation process, which reduces the computational cost only to $\mathcal{O}(M^2) = \binom{M}{2} + \frac{(M-1)(M-2)}{2}$. Here we adopt this greedy approach to tackle the reducibility problem, as also done in Ref. [60]. We start with a multiplex network with M layers, and at each step of the algorithm, we compute the complexity $\mathcal{C}$ of all the possible pairs of layers. We call $\overline{D}$ the pair of layers with the maximum value of complexity, and we consider the set of pairs of layers

whose overlap is larger than or equal to that of $\overline{D}$. We then aggregate the pair of layers D of that set yielding the smallest value of complexity $\mathcal{C}(D)$. The aggregation is performed by taking the union of the edges of the two layers. The rationale behind this particular choice is that it allows to aggregate pairs of layers that are indeed very similar (both in terms of complexity and edges in common). Hence, the iteration of this aggregation procedure will result in a sequence of compressed multiplex networks with $\{M, M-1, M-2, \ldots, 2, 1\}$ layers. Among those M reduced multiplex networks, we choose the one yielding the largest value of $q(\bullet)$.

At first glance, an alternative approach in the aggregation process would be to select the pair of layers that "just" maximise [minimise] the quality function $q(\bullet)$ [the complexity $\mathcal{C}(\bullet)$], without considering the overlap. Yet, both these alternative choices would inevitably create structural artefacts during the aggregation procedure. This is due to the non-monotonic behaviour of the complexity function which, by construction and as shown in Fig. 3.4, assigns low complexity values to the two extreme conditions, i.e. very redundant or very dissimilar pairs of layers. Thus, when maximising (resp. minimising) only one of the two functions, the aggregation procedure might first select a pair of very dissimilar layers (i.e., the two layers share only few edges in common) over couples of layers characterised by a high level of redundancy (i.e. $o \approx 1$) and with (slightly) higher values of complexity.

3.4.1 Reducibility of synthetic multiplex networks

We start testing our method by considering ad-hoc synthetic multiplex networks where some of the M layers are identical by construction. We expect that a good procedure should identify a (compressed) multiplex representation made only of truly distinct layers, in principle. In Fig. 3.8 we report the results of the greedy reduction on four different synthetic benchmarks. In particular, we plot the global quality function $q(\bullet)$ and the complexity $\mathcal{C}(\bullet)$ as a function of the number of layers M (left panels), and the dendrogram corresponding to the greedy aggregation

Benchmark ID	M	$M_{distinct}$	*pattern*	$M_{opt}(C)$	$M_{opt}(VN)$
1	15	5	{3,3,3,3,3}	5	5
2	20	10	{2,2,2,2,2,2,2,2,2,2}	10	12
3	15	5	{3,2,1,4,5}	5	5
4	50	5	{10,10,10,10,10}	5	14
5	10	4	{5,2,2,1}	4	5
6	10	6	{2,2,1,2,2,1}	6	6
7	30	2	{15,15}	2	16
8	35	1	{35}	1	NA

Table 3.3. ***Reducibility of synthetic multiplex benchmarks with ground truth.*** *From left to right, we report the total number of layers M, the total number of distinct layers $M_{distinct}$, the pattern of identical layers in the multiplex, and the number of optimal layers $M_{opt}(C)$ obtained when maximising the quality function $q(\bullet)$. The last column indicates the optimal number of layers $M_{opt}(VN)$ when using the multiplex structural reducibility based on the von Neumann entropy. Notice that in all the cases, $\mathcal{C}(\mathcal{M})$ correctly identifies both the correct sequence of aggregation steps and the best partition, outperforming the procedure based on the von Neumann entropy.*

steps (right panels).

In all the synthetic cases considered, the maximum of the quality function $q(\bullet)$ correctly identifies the partition made of truly distinct layers, while in general the raw value of complexity $\mathcal{C}(\bullet)$ fails to identify the correct partition. This confirms our intuition that, by taking into account

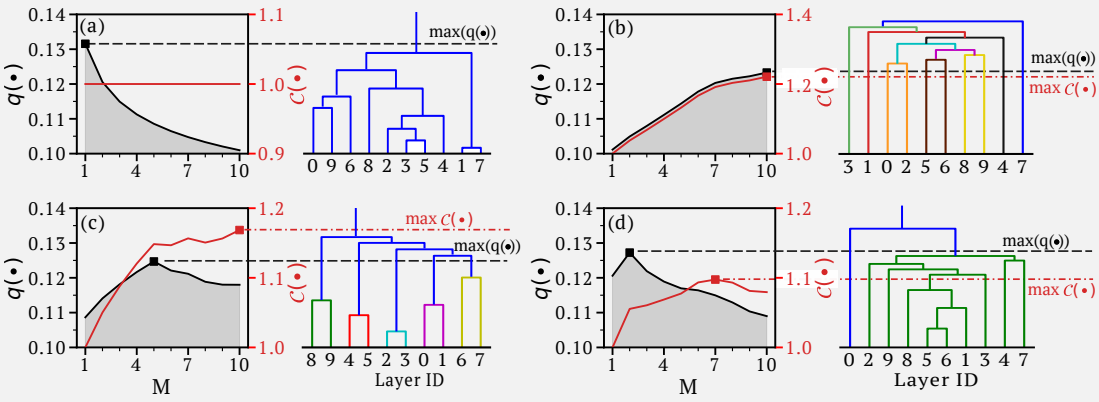

Figure 3.8. ***Reducibility of four synthetic multiplex networks with $N = 1000$ nodes, $\langle k \rangle = 4$ and $M = 10$ layers.*** *The four panels correspond to: (a) ten identical layers; (b) ten distinct layers; (c) five pairs of identical layers; (d) 9 identical layers and 1 distinct layer. Notice that all the distinct layers are independent realisations of Erdös-Rényi random graphs. For each synthetic multiplex we show both the global quality function $q(\bullet)$ and the complexity $\mathcal{C}(\bullet)$ as a function of the number of layers M (left panel) and the dendrogram resulting from the greedy aggregation steps of the reducibility procedure (right panel). Notice that the maximum of the quality function $q(\bullet)$ corresponds to the true partition of layers, while the complexity $\mathcal{C}$ generally fails to identify the correct partition.*

differences in the total number of edges of the multiplex, the quality function $q(\bullet)$ does a better job at discriminating between essential and redundant information. Indeed, while the multiplex with $M = 10$ identical layers is always aggregated into a single-layer graph [Fig. 3.8(a)], in the multiplex with all distinct layers, the maximum of $q(\bullet)$ is attained by the initial configuration with ten layers [Fig. 3.8(b)].

Table 3.3 shows the results of our reducibility procedure for 8 other synthetic benchmarks with variable number of layers, highlighting the pattern of identical layers in each multiplex (i.e. the ground truth). For instance, the pattern $\{3, 3, 3, 3, 3\}$ in the first benchmark corresponds to a multiplex with $M = 15$ layers where every subsequent triplets of layers are identical, i.e., layer1 is identical to layer2 and layer3, layer4 is identical to layer5 and layer6, and so on. We report in the third column of the table the number M distinct of layers that are truly distinct, in the sense mentioned above. For the sake of completeness, we also report the results obtained when considering the reducibility procedure based on the Von Neumann entropy, mentioned in the previous chapter and originally developed in Ref. [60]. These synthetic benchmarks reveal that the reducibility procedure based on complexity correctly identifies all the reduced multiplex networks made of only distinct layers, while the procedure based on the Von Neumann entropy fails in some instances.

To further compare the two methods, we also consider a different set of synthetic multiplex networks, where the average degree and the topology of each layer are not kept fixed. In particular, we first consider two benchmarks with $N = 1000$ nodes, constructed through two different models, namely, Barabasi-Albert linear preferential attachment graphs (BA) [190], and ER graphs. We consider a multiplex consisting of $M = 30$ layers, where every group of 10 layers is identical to a BA graph with $m = 3$, an ER graph with $\langle k \rangle = 4$, and an ER graph with $\langle k \rangle = 8$. Within this setting, the number of distinct layers is by construction equal to 3. In the second benchmark, the number of total layers is equal to $M = 35$. In this case, we vary the average degree for both the ER and BA models. The first five layers correspond to a BA with $m = 3$,

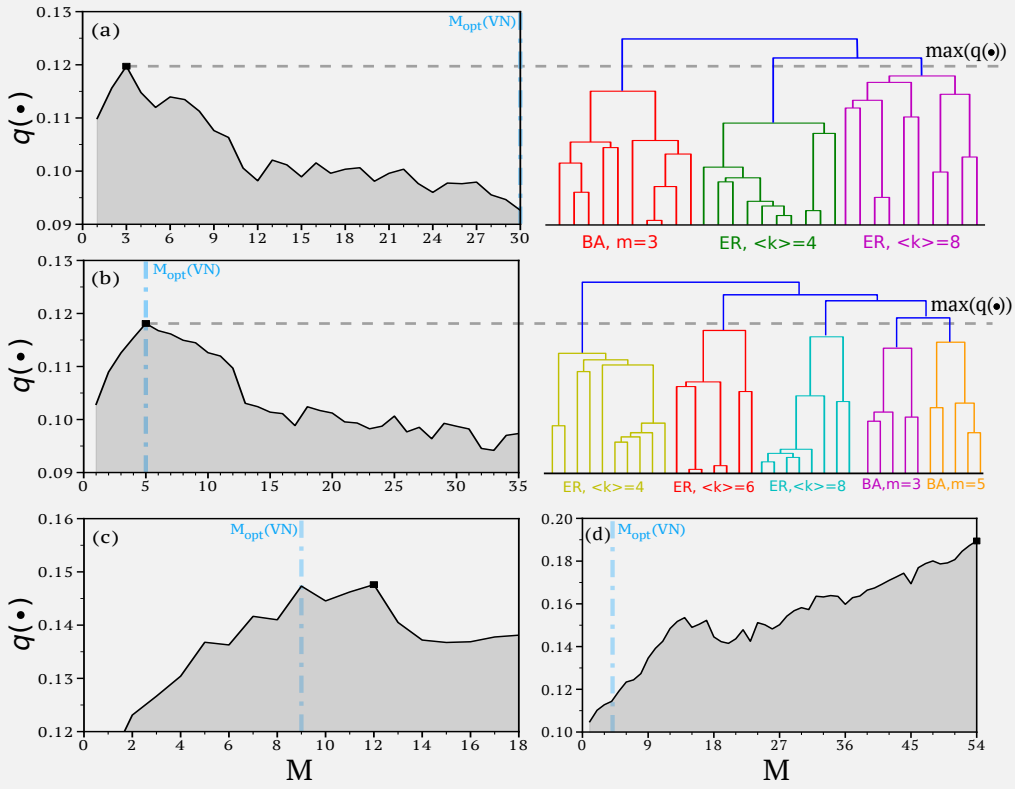

Figure 3.9. ***Reducibility analysis for multiplex networks with layers having different topologies and average degree*** $\langle k \rangle$. *The multiplex networks correspond to: (a) 3 groups of ten layers respectively created with the BA model* $m = 3$, *and ER with* $\langle k \rangle = 4, 8$; *(b) 35 layers with layers created from the BA model* $m = 3, 5$, *and ER with* $\langle k \rangle = 4, 6, 8$. *In panel (c-d), the multiplex consists of 18 (54) layers, which are made of group of layers constructed from the BA, ER, WS models respectively. Each of these groups is composed by 6 layers, one of which is a realisation of a network model (e.g. BA), while the other 5 layers have an increasing amount of edge overlap with the network model realisation, 10%, 25%, 50%, 75%, 100% respectively. For each synthetic multiplex we show the global quality function* $q(\bullet)$ *as a function of the number of layers* M *and the optimal number of layer* $M_{opt}(VN)$ *identified by the method based on the von Neumann entropy (dotted aqua line). Groups of layers associated with the same network model are normally merged first. It seems that the method based on complexity is more conservative than the one based on the von Neumann entropy, especially when considering the multiplex networks with variable level of overlap.*

and the next five layers are identical to a single realisation of a BA model with $m = 5$. The remaining layers are grouped in 7, 6, and 10 layers respectively, which are sampled from an ER graphs with $\langle k \rangle = 4, 6, 8$. Thus, the number of truly distinct layers are only 5. The last two benchmarks are similar to the one introduced in Ref. [60] (Supplementary Note 1), with layers drawn from BA with $m = 4$, ER with $p = 0.05$ and Watts-Strogatz small-world models (WS, $m = 5, p = 0.2$). More precisely, the two benchmarks have $N = 200$ nodes with one (three)

realisations for each model, named "master layers" and called BA 1 (BA 2, BA 3), ER 1 (ER 2, ER 3), and WS 1 (WS 2, WS 3) [63] respectively. For each realisation, we constructed 5 more layers, each characterised by an increasing amount of edge intersection with the corresponding master layer, i.e 10%, 25%, 50%, 75%, 100%. Thus, we obtained two synthetic benchmarks made of 3 (9) groups of 6 layers each, for a total of 18 and 54 layers respectively.

The results of the reducibility process for these four benchmarks are presented in Fig. 3.9. Also in this case, we highlight the behaviour of the global quality function $q(\bullet)$ as a function of the number of layers M, and for two of the benchmarks we report the dendrogram corresponding to the greedy aggregation steps. The aggregation procedure of our method first merges layers generated by the same network model, and then clusters together all the others [see for instance Fig. 3.9 (a-b) right panel]. For comparison, we also highlight the optimal number of layer $M_{opt}(VN)$ identified by the method based on the von Neumann entropy (dotted aqua line). It appears that our method is more conservative with respect to the other one.

3.4.2 Reducibility of real-world multiplex networks

After having checked that $q(\bullet)$ identifies meaningful layer partitions in synthetic multiplex networks, we extended our analysis to real-world multiplex data sets. The results are reported in Table 3.4. Notice that most of the technological and biological multiplex networks in the Table admit reduced representations, which have only a slightly smaller number of layers than the original systems. This is in agreement with the observation that in technological systems structural redundancy is purposedly avoided. Similarly, the poor redundancy observed in biological multiplex networks is in line with the functionally different role played by each layer (protein interaction, functional dependence, mechanical interaction, and so on). However, technological systems exhibit consistently larger values of multiplex complexity than biological systems. A comparison with the reducibility algorithm based on the von Neumann entropy shows again that

Multiplex	M	o	$M_{opt}(C)$	$[max\, q(\cdot)_C]$	$M_{opt}(VN)$	$q(\cdot)_{VN}$
London Tube [60]	13	0.006810	11	0.183	2	0.499
Barcelona Tube [2]	11	0.002367	11	0.224	11	0.513
Bejing Tube [2]	17	0.000197	15	0.199	17	0.528
Berlin Tube [2]	9	0.001359	8	0.214	9	0.461
Airports North America [224]	143	0.003958	129	0.143	93	0.697
Airports Europe [224]	175	0.003185	163	0.162	109	0.675
Airports Asia [224]	213	0.005477	209	0.180	146	0.291
Airports South America [224]	58	0.014244	53	0.187	41	0.682
Airports Oceania [224]	37	0.014532	27	0.185	31	0.665
Airports Africa [224]	84	0.006876	74	0.191	65	0.719
EU airlines [56]	37	0.005964	37	0.151	37	0.411
Train UK [1]	41	0.002687	24	0.120	15	0.225
APS countries [2]	10	0.451138	10	0.176	2	0.047
Aarhus network	5	0.189093	5	0.201	2	0.158
Terrorist network [224]	4	0.153558	4	0.171	2	0.239
Pierre Auger collab. [247]	16	0.006901	10	0.117	15	0.423
Arabidopsis [60]	7	0.007690	6	0.105	7	0.421
Candida [60]	7	0.007892	5	0.177	3	0.620
Celegans [60]	6	0.003095	6	0.114	5	0.430
Drosophila [60]	7	0.004389	5	0.098	6	0.379
Gallus [60]	6	0.012923	5	0.179	5	0.577
Human Herpes-4 [60]	4	0.042056	2	0.196	4	0.353
Human HIV-1 [60]	5	0.022294	5	0.150	4	0.353
Mus [60]	7	0.010776	7	0.106	6	0.375
Oryctolagus [60]	3	0.019231	3	0.209	2	0.500
Plasmodium [60]	3	0.000206	2	0.128	3	0.611
Rattus [60]	6	0.012401	6	0.126	5	0.472
S. Cerevisiae [60]	7	0.017603	5	0.092	3	0.135
S. Pombe [60]	7	0.007070	5	0.099	2	0.206
Xenopus [60]	5	0.025692	5	0.169	4	0.410

Table 3.4. ***Reducibility of technological, social, and biological multiplex networks.*** *From left to right, the columns report the number of layers in the original system (M), the structural overlap (o), the number of optimal layers ($M_{opt}(C)$) obtained when maximising the quality function $q(\bullet)$, the value $max\ q(\bullet)$, and the optimal value of complexity C_{opt} observed. The last two columns show the optimal number of layers $M_{opt}(VN)$ and the corresponding value of the quality function $q(\bullet)_{VN}$ obtained when using the multiplex structural reducibility procedure introduced in Ref. [60]. Although the two methods yield different results, they share similar features, i.e. technological multiplex networks are less likely to be reduced compared to biological and social systems.*

the reducibility based on complexity is in general more conservative and often yields an optimal partition that has a slightly larger number of layers.

3.4.3 Structural properties of reduced multiplex networks

As discussed earlier, a layer reduction procedure is expected to remove redundancies while maintaining as much information as possible about the original system. However, there is generally no *a priori* guarantee that the reduced multiplex obtained by aggregating some of the layers actually preserves any of the structural or dynamical properties of the original multiplex network to a given level of accuracy. To explore this aspect of layer reduction, we compared the distributions of four structural indicators in the original multiplex networks and in the networks obtained by using the aggregation procedure described above. In Fig. 3.10, we report the Kendall's τ correlation coefficient of the rankings induced by total node degree, node activity, participation coefficient, and node interdependence in both synthetic and real-world multiplex networks.

In almost all the multiplex networks considered, the optimal partition identified by the multiplex complexity preserves most of the structural properties of the original system, as confirmed by the relatively high values of correlation ($\tau > 0.8$). Conversely, the optimal aggregations based on the von Neumann entropy [60] often correspond to relatively lower values of correlation. We argue that this is a very desirable feature of the definition of complexity we have proposed. Indeed, a decrease of structural correlation is a clear indication that aggregation is creating structural artefacts. At the same time, the fact that the configuration found by using $q(\bullet)$ always yields high values of correlation with the original multiplex network confirms that the procedure is removing only truly redundant information, preserving most of the salient properties of the system. This is clearly visible when considering synthetic benchmarks [Fig. 3.10(a-d)], where the optimal partition of the multiplex made of only distinct layers is known by construction. Despite the value of the Kendall's correlation decreases as the number of layers diminishes, the method based on multiplex complexity correctly identifies the optimal partition in all the cases considered. For the numerical computation of the node interdependence λ_i, we rely on a novel method that effectively exploits the prime-weight matrix introduced above. We report in Appendix A.1

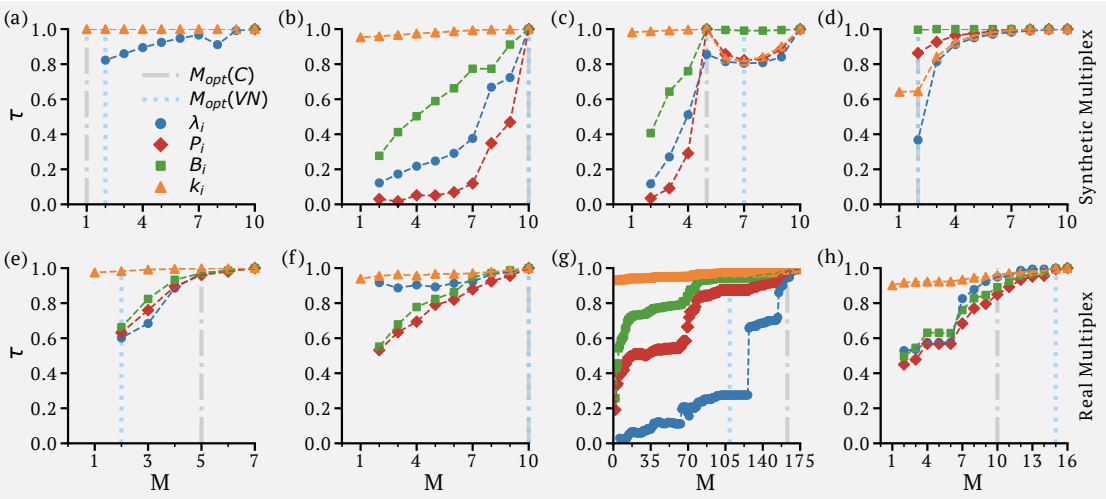

Figure 3.10. ***Impact of reducibility in altering the structural properties of multiplex networks.*** *Kendall's τ correlation between the rankings induced by four structural descriptors in synthetic and real-world multiplex networks and the corresponding reduced networks with M layers. Panels (a)-(d) are for the same synthetic benchmarks shown in Fig. 3.8. The panels on the bottom row correspond to real-world systems, namely (e) Saccharomyces Pombe, (f) APS scientific collaborations among countries, (g) European Airports, and (h) Auger scientific collaboration. The multiplex structural descriptors considered here are total node degree (k_i), node activity (B_i), participation coefficient (P_i), and node interdependence (λ_i). For each network we also indicate the number of layers in the optimal reduced system identified by multiplex complexity ($M_{opt}(C)$, grey dash-dotted line) and by the von Neumann entropy ($M_{opt}(VN)$, aqua dotted line). In all the synthetic cases, the quality function $q(\bullet)$ in Eq. (3.8) correctly identifies the optimal partition as the one where identical layers are aggregated, thus preserving the structural properties of the original multiplex. Conversely, in some cases the partitions found using the von Neumann entropy are different from those expected. Notice that for the benchmark made of only identical layers (a), the correlation coefficient is not defined for two structural descriptors (constant behaviour). For all the real-world networks the multiplex complexity provides a more conservative representation of the system and retains most of the properties of the original graph ($\tau > 0.8$ for all the structural descriptors), while the reduction based on the von Neumann entropy might probably discard important information and/or create structural artefacts.*

the detailed description of the algorithm.

3.4.4 Dynamical properties of reduced multiplex networks

Even though high correlation between the structural properties of a multiplex network and its reduced counterpart indicates that the two systems are structurally similar, this will not guarantee in general that a dynamical process happening on the reduced multiplex network will exhibit a phenomenology similar to that observed on the original multiplex. As an example, we consid-

ered a multiplex susceptible-infected-susceptible (SIS) epidemic model [214] and we computed the epidemic threshold λ_c of the system at each step of the greedy aggregation procedure. In this process, the epidemic threshold depends on the contagion parameter γ/β, which represents the ratio of intra-layer vs inter-layer contagion. In Fig. 3.11, we report the results of our analysis for both synthetic and real-world multiplex networks for different values of the contagion parameter γ/β. Notice that any drop in the value of the critical threshold corresponds to an important change in the structure of the reduced multiplex, e.g., to the formation of new (possibly artificial) structural patterns. This is easily observable in synthetic benchmarks [Fig. 3.11(a)-(d)], where the epidemic threshold of the reduced multiplex remains the same as that of the original multiplex up to the point where $q(\bullet)$ is optimal, and then decreases abruptly. This means that, with respect to epidemic spreading, the reduced multiplex obtained by optimising the quality function $q(\bullet)$ has basically the same dynamical behaviour as the original multiplex. Further aggregations yield a system with different dynamics. As a result, the optimal reduced multiplex network

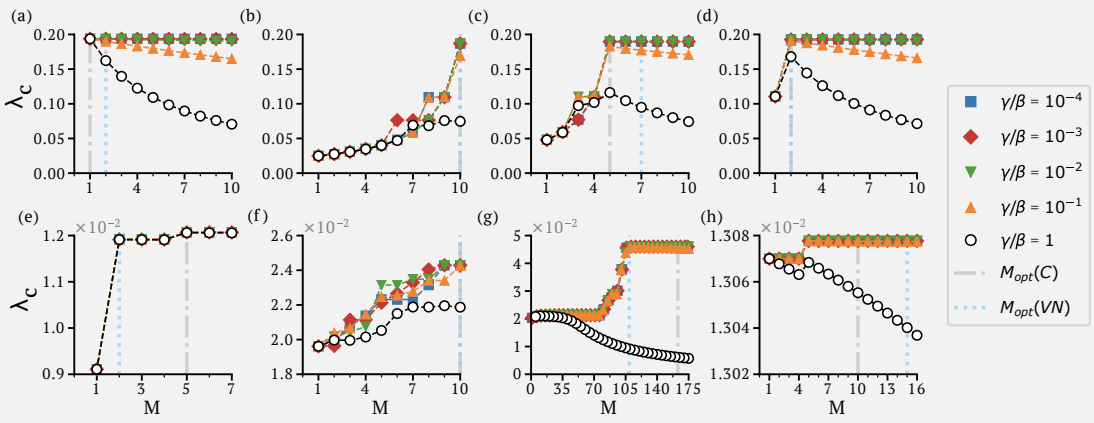

Figure 3.11. ***Impact of reducibility in altering the dynamical properties of multiplex networks.*** *Epidemic threshold λ_c for the multiplex SIS dynamics [214] as a function of the number of layers M and for different values of the contagion parameter γ/β, on the same synthetic and real-world multiplex networks shown in Fig. 3.10. For each network we indicate the number of layers in the optimal reduced system identified by multiplex complexity ($M_{opt}(C)$, grey dash-dotted line) and by the von Neumann entropy ($M_{opt}(VN)$, aqua dotted line) [60]. Interestingly, drops in the epidemic threshold correspond to dramatic changes in the structure of the multiplex, e.g., the formation of new structural patterns. This is clearly observed when considering the synthetic benchmarks (a)-(d), where the drops occur when truly distinct layers are aggregated. The reducibility procedures based on the von Neumann entropy and multiplex complexity provide similar results in real-world networks [panels (e)-(h)], even though they yield different optimal partitions.*

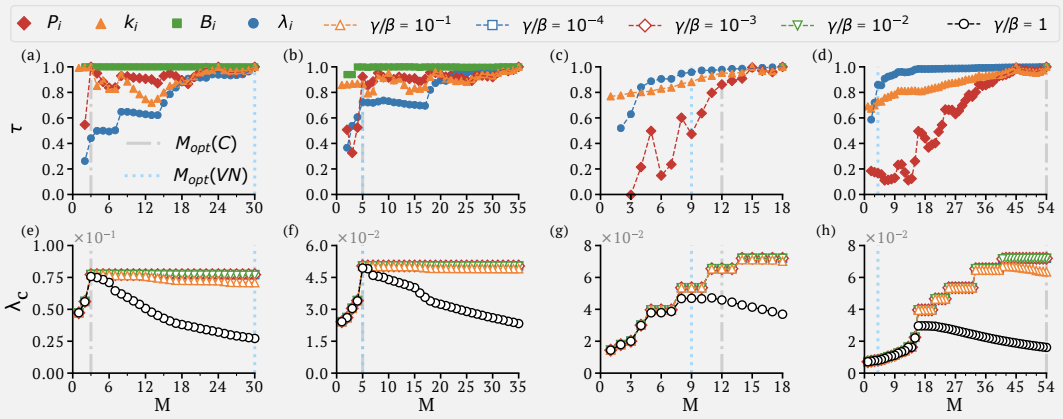

Figure 3.12. ***Impact of reducibility in altering structural and dynamical properties of four synthetic multiplex networks made of layers with different topologies and average degrees.*** *For each of the four synthetic multiplex networks of Fig. 3.9, made of 30, 35, 18 and 54 layers, we report the Kendall's τ correlation of the four structural descriptors (top row) and the critical threshold of the SIS dynamic bottom row) as a function of the greedy aggregation steps of the reducibility procedure. For each network, we also indicate the number of layers in the optimal reduced system identified by multiplex complexity ($M_{opt}(C)$, grey dash-dotted line) and by Von Neumann entropy ($M_{opt}(VN)$, aqua dotted line). Interestingly, the quality function based on Complexity shows a more conservative behaviour compared to the one based on the Von Neumann Entropy. In addition, the greedy aggregation steps for both methods are almost identical, so that couple of layers having the same topology will be aggregated first.*

(which has a smaller number of layers) can be used to make meaningful predictions about the dynamics of spreading of the original system. These results provide further evidence that the multiplex reduction based on Kolmogorov complexity somehow outperforms the reduction based on the von Neumann entropy.

We obtain a similar but more intriguing picture for real-world multiplex networks, as shown in Fig. 3.11(e)-(h). We notice that both methods preserve most of the information of the original multiplex, performing approximately the same. Nevertheless, by looking at both the structural and dynamical features over the aggregation steps, it appears that the method based on $\mathcal{C}(\mathcal{M})$ is a bit more conservative, and finds a reduced multiplex that simultaneously preserves as much as possible of both the structural and dynamical features of the original system. Indeed, the best layer partition identified by the method proposed here has high values of Kendall's correlation of structural properties (≈ 0.8) and small variation of the epidemic threshold

In Fig. 3.12, we report additional results for the synthetic benchmarks with different topologies considered above (i.e., the multiplex networks in Fig. 3.9). In this case, we plot the Kendall's τ correlation distributions of the four structural indicators and the epidemic threshold λ_c of the system at each step of the greedy aggregation procedure.

It is quite remarkable that a reduction of layers based on the multiplex complexity proposed here usually produces reduced graphs that are sensibly different from those obtained using the classical reduction method based on the von Neumann entropy [60]. In particular, it is worth noting that when using the latter method it is not straightforward to pinpoint a specific graph property as responsible for a change of value of the von Neumann entropy (this fact was also highlighted in the Supplemental Material of Ref. [60]). A difference in the placement of a single edge of the graph can frequently result in relatively large fluctuations of the value of the von Neumann entropy. By contrast, the multiplex complexity $\mathcal{C}(\mathcal{M})$ proposed here links quite closely to the traditional meaning of complexity of a system as the amount of information needed to fully describe it (see also the definitions of complexity presented in Chapter 1.2). Moreover, because of the way $\mathcal{C}(\mathcal{M})$ is defined, its value varies in a somehow predictable way if the edges of the graph are reorganized. In particular, if we add a single edge to an existing multiplex, then we can expect the value of multiplex complexity to change only slightly. If the newly added edge increases the structural overlap of the multiplex, then the value of $\mathcal{C}(\mathcal{M})$ will increase if the original multiplex had a small structural overlap, or decrease if the multiplex had a large structural overlap. In this sense, $\mathcal{C}(\mathcal{M})$ is more closely associated with the structure of the system, and its changes provide information that are more readily interpretable.

3.5 Time evolution of complexity

The multiplex complexity $\mathcal{C}(\mathcal{M})$ can be used to track the temporal evolution of the structure of time-varying multiplex networks. In Fig. 3.13, we show how the complexity of five large-

scale multiplex networks has changed over time (see Appendix A.2 for details about the data sets). $\mathcal{C}(\mathcal{M})$ provides an interesting picture of the alternating behaviour of the Internet Movie Database (IMDb) movie co-starring network over about a century [Fig. 3.13(a)], and of the network of 35 major assets in the NYSE and NASDAQ financial markets in the period 1998-2013

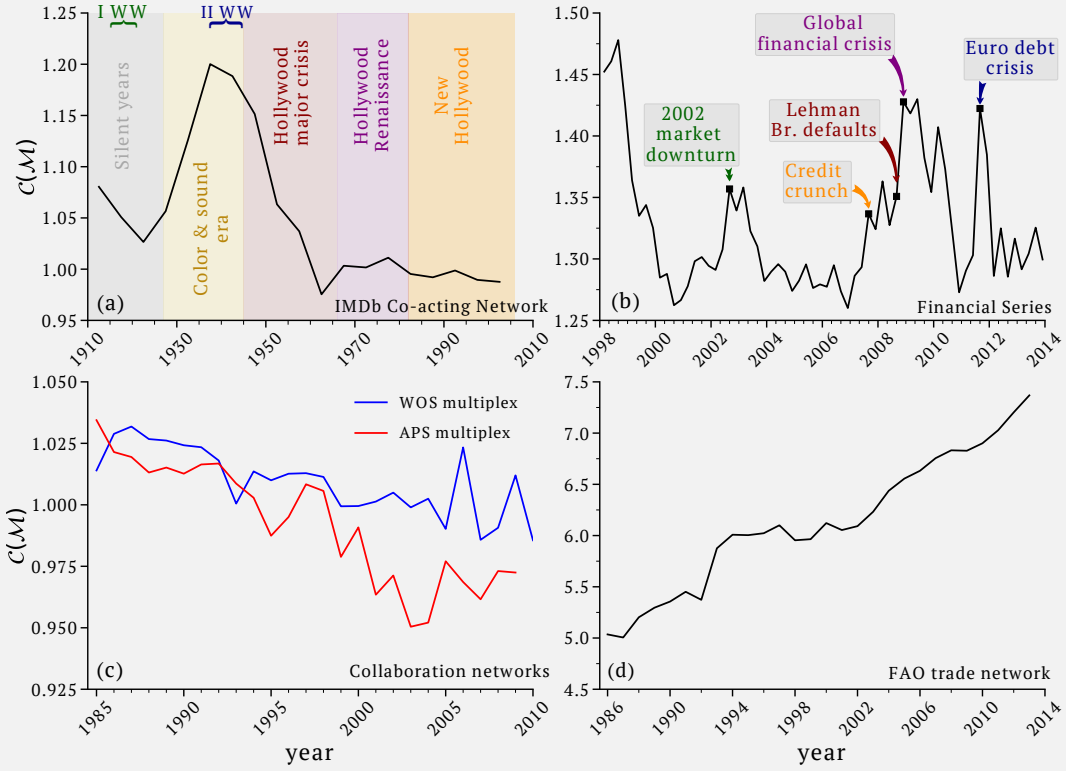

Figure 3.13. ***Multiplex complexity to characterise four different real-world time-varying multiplex networks***. *We report $\mathcal{C}(\mathcal{M})$ as a function of time for four different time-varying multiplex networks, namely, (a) the IMDb co-starring network, (b) the financial multiplex constructed from price time series of 35 major assets in NYSE and NASDAQ, (c) the co-authorship multiplex of collaboration in American Physical Society (APS) journals and Web of Science (WOS), and (d) the FAO food import/export multiplex network. Interestingly, some of the local maxima seem to correspond to some periods of crisis in (a) and (b). In the IMDb data set, the highest value of complexity appears in the period between 1935-1940 (we highlight the middle point, i.e. 1937.5 in the plot), which includes the starting date of WWII. Yet, not all the maxima of $\mathcal{C}(\mathcal{M})$ correspond to important periods of instability or crisis, as it is clear from panel (b), where the peak of complexity in 1999 does not capture any period/event of economic depression. The values of complexity in the physics collaboration multiplex, for both the APS and WOS data sets (c), have remained pretty stable over time, and reveal that those systems indeed benefit only marginally from a multilayer representation. Finally, in the FAO food import/export multiplex network the complexity has been increasing considerably over time (d), reflecting the relevant role played by globalisation in the last twenty years in re-shaping the international food market. More details and additional results on these four data sets are reported in Appendix A.2. Abbreviations: WW—World War; Lehman Br.—Lehman Brothers.*

[Fig. 3.13(b)]. In both cases, some of the local maxima of complexity may be associated with some periods of crisis in each data set, while local minima of complexity seem to be precursors of renaissance in IMDb and of stability in the financial market. Yet, not all the maxima of $\mathcal{C}(\mathcal{M})$ correspond to important periods of instability or crisis, as it is clear from Fig. 3.13(b), where the peak of complexity in 1999 does not reflect any period/event of economic depression. The value of complexity of scientific collaboration networks [APS and Web of Science, Fig. 3.13(c)] has remained stable around $\mathcal{C}(\mathcal{M}) = 1$ over the last 35 years. This is mainly due to the fact that in these multiplex networks each layer represents a different field or sub-field of science, and authors normally tend to publish in one or at most a couple of fields. In fact, the structural overlap of those multiplexes is always very small, and the majority of pairs of nodes are connected in at most two layers. As a result, there is not much benefit in considering the multiplex representation, since the information encoded in the different layers is comparable to that contained in the corresponding aggregated graph.

The complexity of the Food and Agriculture Organization of the United Nations (FAO) multiplex network of food exchange has been increasing steadily in the last 30 years [Fig. 3.13(d)]. This is most probably linked to the globalisation of commercial exchanges, which is also reflected in a more intricate pattern of relations among countries across a wide range of products.

3.6 Planar embedding of real-world multiplex networks

In Fig. 3.14 we show how multiplex complexity can be used to obtain a planar embedding of multiplex networks of a different kind, and to reveal the presence of interesting clusters. For each multiplex network, we used the maximum value of the quality function $\max q(\bullet)$ as one of the coordinates, and the normalised epidemic threshold $\tilde{\lambda}_c = \lambda_c / M_{opt}(\mathcal{C})$ of the SIS dynamics [214] with contagion parameter $\gamma/\beta = 1$ as the other one. Notice that $\tilde{\lambda}_c$ removes the dependence on the number of layers of the multiplex, making it possible to compare reduced multiplex networks

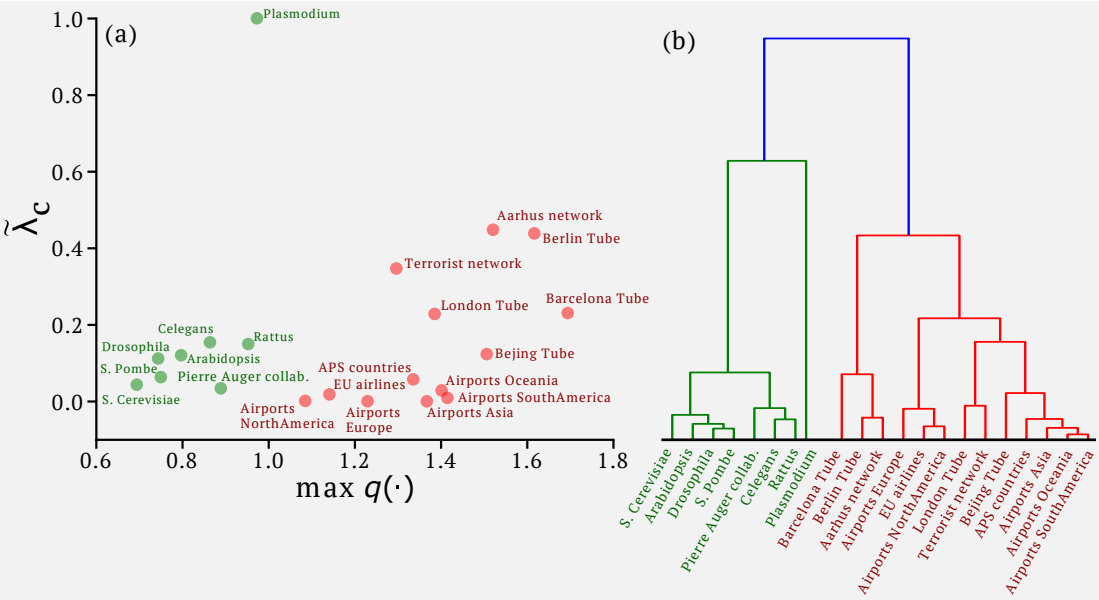

Figure 3.14. ***Multiplex cartography in real networks by means of orthogonal multiplex metrics.*** *We map real-world systems in the plane* $(\max q(\bullet) - \tilde{\lambda}_c)$ *(a) and we report the corresponding dendrogram obtained through Ward's hierarchical agglomerative clustering (b). It appears that these two orthogonal descriptors alone are sufficient to identify two large classes of real-world multiplex networks, namely, biological networks on the one side (green cluster) and techno-social systems on the other side (red cluster).*

with a different number of layers. Moreover, since the epidemic threshold is intimately connected to the spreading dynamics on a graph, the information it provides is somehow orthogonal to that captured by multiplex complexity, which is instead a purely structural quantity. In Fig. 3.14(a), we indicated with different colours the two largest groups obtained through hierarchical clustering in the $[\max q(\bullet), \tilde{\lambda}_c]$ plane, while in Fig. 3.14(b), we show the corresponding dendrogram, highlighting all the aggregation steps, where at each step of the procedure we merge two clusters based on the minimum increase in total within-cluster variance over all possible pairs [263]. These two descriptors are already sufficient to cluster multiplex networks with different functions, so that all the biological multiplex networks appear in the same cluster and social and technological systems are put in another cluster. An even more intriguing picture, where biological, social, and technological networks are put in three distinct clusters, is obtained when the normalised epidemic threshold is replaced by another dynamical descriptor, i.e., the maximal entropy rate per node h_{max} [264]. Again, in this case the two orthogonal measures (i.e. a structural and

dynamical metric) are sufficient to cluster systems in macro-families. Note that $h_{\max}$ is linked to the dispersiveness of random walks on a graph, and it thus carries information about the large-scale dynamical properties of a multiplex [264]. For a multiplex $\mathcal{M}$, the maximal entropy rate is defined as:

$$h_{max} = \log \lambda_{max} \tag{3.9}$$

where λ_{max} is the maximum eigenvalue of the overlapping matrix $O = \{o_{ij}\}$ associated with $\mathcal{M}$ [264]. To account for the dependence of λ_{max} on the total number of nodes in the graph, we use the normalised maximal entropy rate:

$$\tilde{h}_{\max} = \frac{h_{\max}}{N}. \tag{3.10}$$

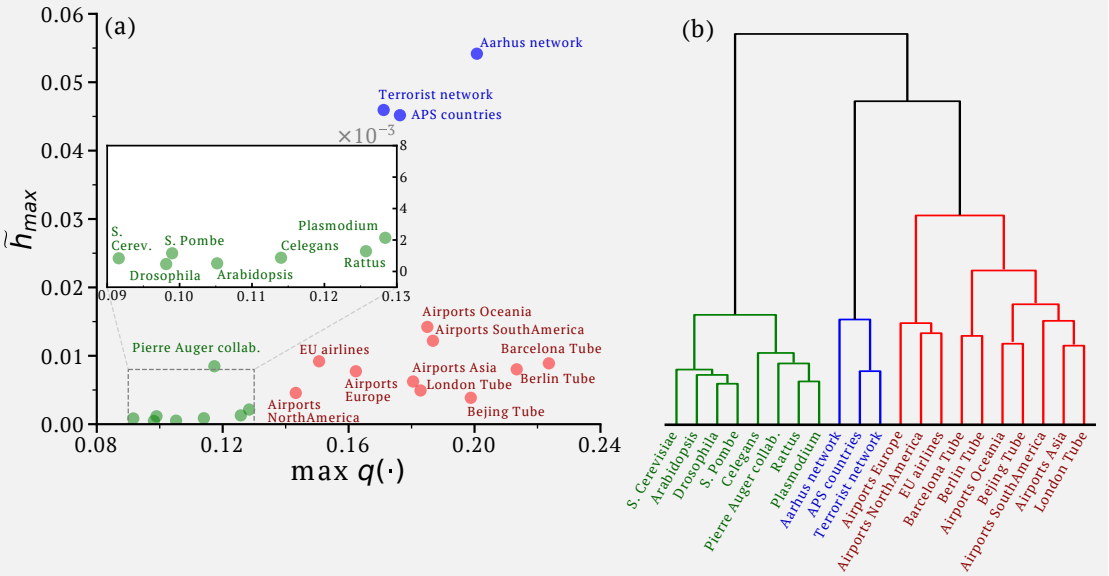

Figure 3.15. ***Cartography of real-world multiplex networks.*** *We report the multiplex cartography of real-world systems in the plane* $(\max q(\bullet), \tilde{h}_{\max})$ *(a) and the corresponding dendrogram obtained through Ward's method hierarchical agglomerative clustering. It looks like these two structural descriptors alone are able to identify three classes of real-world multiplex networks, namely, biological networks (green cluster), social systems (blue cluster), technological systems (red cluster).*

In Fig. 3.15, we indicated with different colours the three largest group obtained through Ward's method [263] in the [$\max q(\bullet)$, $\tilde{h}_{\max}$] plane, while in Fig. 3.15(b), we show the corresponding dendrogram, highlighting all the aggregation steps.

Part II

Optimisation in multiplex networks

Chapter 4

Optimisation and Pareto Optimality

In this chapter we provide a brief overview of single and multi-objective optimisation theory. We then formally introduce the concept of Pareto optimality with explanatory examples. Finally, we review some of the performance metrics proposed in the literature to compare quantitatively the solutions of different multi-objective optimisation problems.

4.1 Background

Imagine you need to buy a car; you look at the market and consider all the possible options. Cars are available on the market with a range of different prices, from a few thousand to a few hundred thousand pounds. They differ from each other by a wide set of characteristics, such as global comfort, maximum speed, stability, road-holding, number of seats, and of course the brand. For our purpose, let us consider only one of these features, namely, global comfort. Our aim is to

select, among all the cars available on the market (i.e. all the *feasible solutions*), the best one that simultaneously maximises comfort and minimises cost (i.e the two *objective functions*) [265]. For the sake of simplicity, let us suppose that all cars are within a fixed range of price (i.e. five to fifty thousand pounds), and degree of comfort (i.e. from 40 to 90%), as shown in Fig. 4.1. How do we select a solution between all the possible available cars? For sure, if we compare solution A with respect to X, solution A is better than X, since it has the same level of comfort at a cheaper price. A similar argument can be used when comparing solution B with X (i.e. at the same cost, solution B offers a higher comfort). Yet, when comparing solution A and B, we cannot determine which one is better than the other. This is because, an increase in the degree of comfort comes at the expense of the cost. A similar argument is valid for any pair of red points in Fig. 4.1. In other words, these points represent those solutions for which no improvement can be achieved in one objective function without hindering the other one. As a result, all these "trade-off" solutions can be considered as "optimal" with respect to the problem of minimising the cost and maximising the comfort. But in practice, how do we select one of these optimal

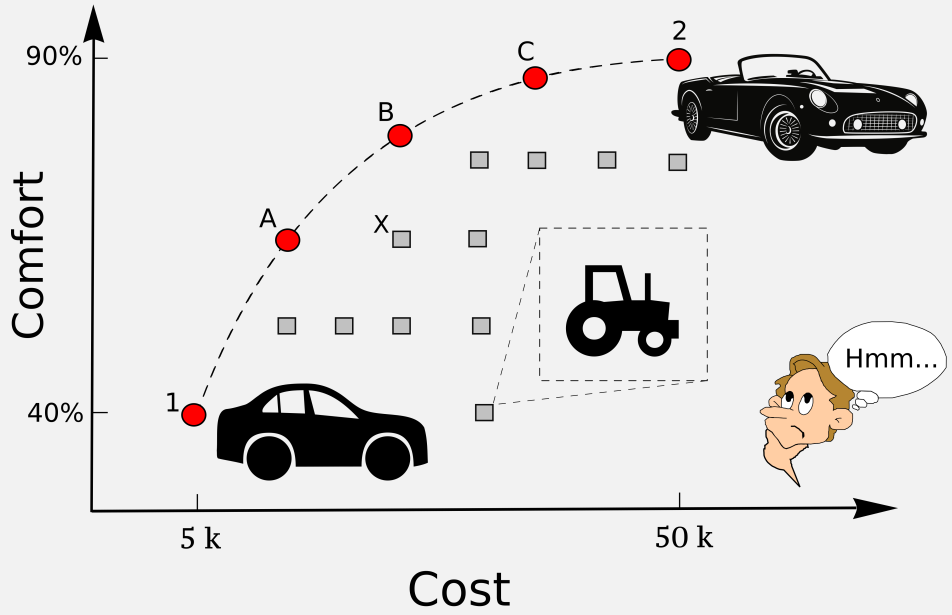

Figure 4.1. ***Solutions of a multi-objective optimisation problem***. *Hypothetical example in a car-buying decision making problem. Trade-off solutions are denoted in red for this multi-objective optimisation problem. Adapted from [265].*

solutions? There is not a single answer. For example, a wealthy buyer, for whom comfort is the primary goal, would select solution 2, since it corresponds to the highest level of comfort. By contrast, if the budget is very limited, the buyer would be inclined to choose solution 1.

In more formal terms, the example presented above falls in the class of optimisation problems. Optimisation can be regarded as the procedure of finding and comparing feasible solutions until no further improvement can be found [265]. Solutions are considered good or bad in terms of an *objective function*, which often reflects the efficiency or quality of a product, the cost of production, a product's reliability, or other factors [265]. In the last decades, a large amount of research and applications in the field of optimisation have mainly focused on considering single-objective optimisation (SOO), i.e. the problem that involves the optimisation of only one objective function. Hence, the existence of several kinds of single-objective optimisation algorithms [266]. Some of them, for instance, are based on deterministic search principles, while others on stochastic search principles, which allow optimisation algorithms to find globally optimal solutions in a more reliable way [265]. Examples of these principles include gradient-based and heuristic-based search techniques.

However, as we have seen from the example above, most real-world problems cannot be framed in the optimisation of a single-objective function. On the contrary, they often involve more than one objective function at a time. The optimisation problem dealing with more than one objective function at a time is known as *multi-objective optimisation*. Note that the multiple objectives involved in a multi-objective optimisation process are often in conflict with one another so that the optimisation problem becomes challenging. For instance, in the car example, an increase in the degree of comfort, always comes with an increase in the price. More importantly, as seen above, since no solution can be considered as an optimum solution to all the multiple conflicting objectives, the resulting multi-objective optimisation problem transforms to finding a number of different trade-off optimal solutions. In particular, a solution that optimises with respect to one of the objective functions often requires a compromise in other objective functions. As a result,

without additional information, none of these trade-offs can be considered better than the others.

In the following, we will formally introduce the concept of a multi-objective optimisation problem and mention some of the techniques envisaged in the literature to select one of the trade-off solutions.

4.2 Multi-objective Optimisation Problem

To define the optimal solutions for a multi-objective optimisation problem, we need to introduce an order relation on a generic set A.

Let us introduce the basic ingredients from set theory:

Definition 4.2.1 (Binary relation)

A **binary relation** R *on two sets* A *and* B *is a subset of the Cartesian product* $A \times B$.

If $(a, b) \in R$, *we sometimes write* $a\,R\,b$.

In particular, when we say that R is a binary relation on a set A, we mean that R is a subset of $A \times A$.

Definition 4.2.2 (Strict partial order)

Let R *be a binary relation on a generic set* A. *We say that such relation is a* **strict partial order** *on* A *if holds:*

1. irreflexive: *if* $(x, x) \notin R$ *for every* $x \in A$;
2. transitive: *if* $(x, y) \in R$, $(y, z) \in R$ *imply* $(x, z) \in R$.

We will denote binary relations with such properties with $\prec$. In particular, we can now define the (usual) partial order, denoted with the symbol $\preccurlyeq$, as:

Definition 4.2.3 (Partial order)

Let R be a binary relation on A. We say that it is a **partial order** *iff:*

1. reflexive: *if $x \preccurlyeq x$ for every $x \in A$;*
2. antisymmetric: *if $x \preccurlyeq y$ and $y \preccurlyeq x$ imply $x = y$;*
3. transitive: *if $x \preccurlyeq y$ and $y \preccurlyeq z$ imply $x \preccurlyeq z$.*

If a partial order holds the property of being *complete* (i.e., if $a \preccurlyeq b$ or $b \preccurlyeq a$ for $a, b \in A$), then it is called a *total order*.

Definition 4.2.4 (Multi-objective optimisation problem)

We state the **multi-objective optimisation problem** *(MOOP) in the form defined in Ref. [267]:*

$$\begin{aligned} &\textit{minimise} \quad \{f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_k(\mathbf{x})\} \\ &\textit{subject to} \quad \mathbf{x} \in S \end{aligned} \tag{4.1}$$

where we have k (≥ 2) objective functions $f_i : \mathbb{R}^n \to \mathbb{R}$.

We indicate by $\mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_k(\mathbf{x}))^T$ the vector of objective functions. Note that the decision vector $\mathbf{x} = (x_1, x_2, \ldots, x_n)^T$ belongs to the *feasible region* S (also referred as *search space*), which is a subset of the *decision variable space* $\mathbb{R}^n$, i.e. $S \subseteq \mathbb{R}^n$, where n is the dimensionality of the decision variable space. We avoid fixing the form of the *constraint functions* forming S to be more general in the discussion. A candidate solution $\mathbf{x}$ that does not satisfy all the constraint functions forming S [i.e. $\mathbf{x} \notin S$] is called *infeasible solution*. By contrast, if the candidate solution $\mathbf{x} \in S$, then it is called a *feasible solution*.

It is worth stressing that the word 'minimise' refers to the problem in which we want to minimise all the objective functions simultaneously. However, we deliberately neglect the case where no conflict between the objective functions exists. In fact, in this case, the optimal solution

can be easily found by minimising each objective function individually. Here, we refer to the definition of conflicting objectives given by Deb in Ref. [265]. He defines a set of objectives as conflicting if there is no solution that simultaneously achieves for each objective the optimal value; otherwise the set is non-conflicting. In the following, we will always assume that there does not exist a single solution that is optimal with respect to every objective function. As a result, the objective functions will be at least partially conflicting.

Notice also that it is not restrictive to assume only the minimisation of each objective function in the multi-objective problem defined in Eq. (4.1). In fact, the duality principle [268, 269] allows us to convert a minimisation problem into a maximisation problem. In other words, $\max\{\mathbf{f}(x)\} = -\min\{-\mathbf{f}(x)\}$ and $\arg\max\{\mathbf{f}(x)\} = \arg\min\{-\mathbf{f}(x)\}$.
In the following, we adopt the same notation used in Ref. [267]. We denote the image of the feasible region by $\mathcal{Z}$ $(= \mathbf{f}(S))$ and call it a *feasible objective region*. It is a subset of the *objective space* $\mathbb{R}^k$. The elements of $\mathcal{Z}$ are called *objective vectors* or *criterion vectors* and denoted by $\mathbf{f}(\mathbf{x}) = \mathbf{z} = (z_1, z_2, \ldots, z_k)^T$, where $z_i = f_i(\mathbf{x})$ for all $i = 1, \ldots, k$ are the *objective values*.

4.2.1 Pareto optimality

In this section, we review the concept of optimality, which is essential in optimisation. In the context of single-objective optimisation, the concept of optimality is rather easy to define since the set of real numbers equipped with the usual "less or equal than" $(\leq)$ or "greater or equal than" $(\geq)$ relations is totally ordered. By contrast, in a MOOP the objective space is only partially ordered, hence there is no natural ordering (e.g. there is no meaningful relation that allows us to rank the vectors $(1, 2)$ and $(2, 1)$). As previously discussed in the car-buying example, we can intuitively say that a candidate solution is *Pareto-optimal* if there are no other solutions which hold an improvement with respect to each one of the objective functions.

To introduce the formal concept of *Pareto-optimality*, let us first explore the *dominance strict*

partial order in the k-dimensional objective space, which naturally induces a relationship on the domain space of the problem [265, 267, 270]:

Definition 4.2.5 (Dominance)

Let $\mathbf{x} \equiv (x_1, \ldots, x_k)$ *and* $\mathbf{y} \equiv (y_1, \ldots, y_k) \in \mathbb{R}^k$ *be two* k*-dimensional vectors,* $k \geq 1$*. We say that* $\mathbf{x}$ dominates $\mathbf{y}$ *(*$\mathbf{x} \prec \mathbf{y}$*) if* $x_i \leq y_i$ *for* $i = 1, 2, \ldots, k$ *and* $\exists\, j \in \{1, 2, \ldots, k\}$ *such that* $x_j < y_j$*.*

For $k = 1$ (i.e. single-objective optimisation), the definition above is trivially satisfied by the total order on $\mathbb{R}$. The *dominance* relationship allows us to give the formal definition of Pareto-optimality:

Definition 4.2.6 (Pareto-optimality)

Let $\mathbf{x}$ *be a* feasible *point for problem (4.1). We define* $\mathbf{x}$ *as* Pareto-optimal *point for the problem (4.1) if* $\nexists\, \mathbf{y} \in S$ *such that* $\mathbf{f}(\mathbf{y}) \equiv (f_1(\mathbf{y}), \ldots, f_k(\mathbf{y}))$ dominates $\mathbf{f}(\mathbf{x}) \equiv (f_1(\mathbf{x}), \ldots, f_k(\mathbf{x}))$.

Now, we provide the definition of *Pareto front* which is the set formed by the Pareto-optimal feasible points for problem defined in Eq. (4.1).

Definition 4.2.7 (Pareto front)

The Pareto front*,* $\mathcal{PF}$*, of problem (4.1) is the set of all the* Pareto-optimal *feasible points, which are also called as non-dominated points or non-dominated set [265, 267].*

Later in this thesis, when a problem as in Eq. 4.1 will be solved using an heuristic algorithm, we shall refer to *theoretical* Pareto-optimality and *theoretical* Pareto front (or alternatively *approximated* Pareto front) to distinguish it from the one obtained from the real-world data, i.e. the *observed* Pareto front. It is worth mentioning that the concept of Pareto-optimality was first considered in the context of economic equilibrium and welfare theories at the beginning of the 20th century. The term is named after Vilfredo Pareto, even though the first rigorous mathematical treatment was given by Kuhn-Tucker in 1951 [271]. Further contributions are given in [272].

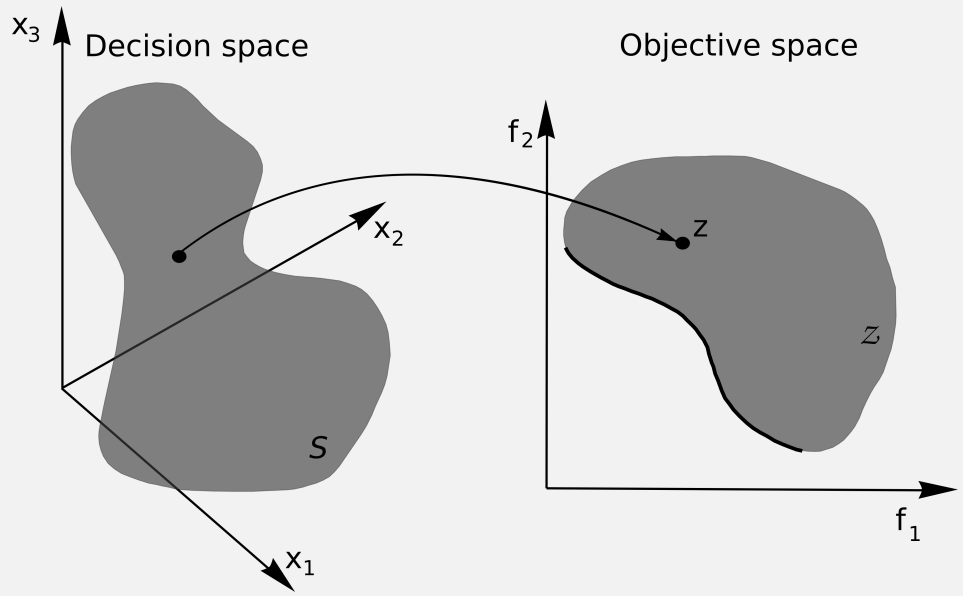

Figure 4.2. ***Illustration of a feasible region and objective space of MOOP***. *Representation of the decision variable space and the corresponding objective space along with the Pareto-optimal solutions, marked with a continuous line. Reprinted from [265, 278].*

Nowadays, this concept is widely used in different contexts, including engineering applications [273–275], stochastic games [276], biology [80, 81, 270, 277], and neuroscience [82, 83].

Note also that, in practical terms, the time complexity for computing the Pareto front in a general MOOP with N feasible solutions is $\mathcal{O}((N \log N)^{k-2})$ for $k \geq 4$, while $\mathcal{O}(N \log N)$ for $k \leq 3$ [279]. We shall reuse this information in Chapter 6 when applying Pareto-optimal principles in the context of multiplex percolation. For the sake of clarity, we report in Figure 4.2 a schematic representation of a feasible region $S \subset \mathbb{R}^3$ and its image, i.e. a feasible objective region $\mathcal{Z} \subset \mathbb{R}^2$, together with the Pareto front curve (solid line).

In several MOOPs, especially in the presence of noise, the concept of Pareto-optimality may impose a too severe restriction. As a result, it is useful to rely on the concept of ε-non-dominance [280], which can be seen as a relaxation of the Pareto-optimal definition, or as an *approximated* Pareto-optimality.

Definition 4.2.8 (ε-Dominance)

Let $\varepsilon \geq 0$ be a non negative real number. Let $\mathbf{x} \equiv (x_1, \ldots, x_k)$ and $\mathbf{y} \equiv (y_1, \ldots, y_k) \in \mathbb{R}^k$ be two k-dimensional vector, $k \geq 1$. We say that $\mathbf{x}$ ε-dominates $\mathbf{y}$ *($\mathbf{x} \prec_\varepsilon \mathbf{y}$) if $x_i \leq y_i + \varepsilon$ for $i = 1, 2, \ldots, k$ and $\exists\, j \in \{1, 2, \ldots, k\}$ s.t. $x_j < y_j + \varepsilon$.*

For $\varepsilon = 0$ we obtain the same definition of dominance previously introduced. The definition of *Pareto-ε-optimality* is a generalisation of Def. 4.2.6.

Definition 4.2.9 (Pareto-ε-optimality) *Let $\varepsilon \geq 0$ be a non negative real number, $\mathbf{x}$ be a* feasible *point for problem (4.1). We say that $\mathbf{x}$ is a* Pareto-ε-optimal *point for problem (4.1) if $\nexists\, \mathbf{y} \in S$ such that $F(\mathbf{y}) \equiv (f_1(\mathbf{y}), \ldots, f_k(\mathbf{y}))$* ε-dominates *$F(\mathbf{x}) \equiv (f_1(\mathbf{x}), \ldots, f_k(\mathbf{x}))$.*

Lastly, the *Pareto-ε-front* is defined as:

Definition 4.2.10 (Pareto-ε-front) *Let $\varepsilon \geq 0$, the* Pareto-ε-front*, $\mathcal{PF}_\varepsilon$, of problem (4.1) is the set of all the* Pareto-ε-optimal *feasible points.*

4.2.2 Special solutions

In this section, we define some particular solutions that are often considered in multi-objective optimisation algorithms and that we shall extensively consider in the next chapters.

First, for each of the k conflicting objectives of a MOOP, there exists at least one particular optimal solution that minimises the objective function f_k among all the feasible solutions. As a result, if we consider an objective vector formed by these individual optimal objective values, we obtain the so-called *ideal objective vector*. Here, we report the formal definition of Ref. [265]:

Definition 4.2.11 (Ideal point)

The k-th component of the ***ideal objective vector*** *z^* is the constrained minimum solution*

of the following problem:

$$\begin{aligned} &\textit{Minimise} && f_k(\mathbf{x}) \\ &\textit{Subject to} && \mathbf{x} \in S. \end{aligned} \tag{4.2}$$

Hence, if the minimum solution for the k-th objective function is the decision vector $\boldsymbol{x}^*(k)$ *with function value* f_k^**, the* **ideal vector** *is then defined as follows:*

$$\boldsymbol{z}^* = \boldsymbol{f}^* = (f_1^*, f_2^*, \ldots, f_k^*). \tag{4.3}$$

In other words, the ideal objective vector corresponds to an array of lower bounds of all objective functions and, unsurprisingly, it is usually a non-existent solution. Figure 4.3 provides a pictorial representation of the ideal point $\mathbf{z}^*$ in a two-dimensional objective space $\mathcal{Z}$. For each objective function f_k, there exists at least one solution in the feasible space that shares the same value with the corresponding k-component in the ideal point (i.e. $\mathbf{z}^{*(1)}$ and $\mathbf{z}^{*(2)}$ in Figure 4.3). As we shall see in this chapter, the knowledge of the ideal point can be particularly useful. For example, when we aim to sample one or a few solutions from the non-dominated front and no additional information is available, Pareto-optimal points closer to the ideal one are usually considered to be excellent candidates [267, 278, 281].

Second, when computing an approximated Pareto front for a MOOP, some of the algorithms present in the literature often require a non-existent reference solution that has objective values strictly better than any other solutions in the search space. To this end, we define the utopian objective point as [265]:

Definition 4.2.12 (Utopian point)

The ***utopian objective point*** z^{**} *has each of its components slightly smaller than that of the ideal objective point, or equivalently* $z^{**} = z_i^* - \varepsilon_i$ *with* $\varepsilon_i > 0$ *for all* $i = 1, 2, \ldots, k$.

Finally, the third important objective vector to mention is the so-called *nadir objective vector* z^{nad}, which represents the upper bound of each objective function in the entire Pareto-optimal set (i.e. the non-dominated solutions). The nadir objective vector, however, is completely different

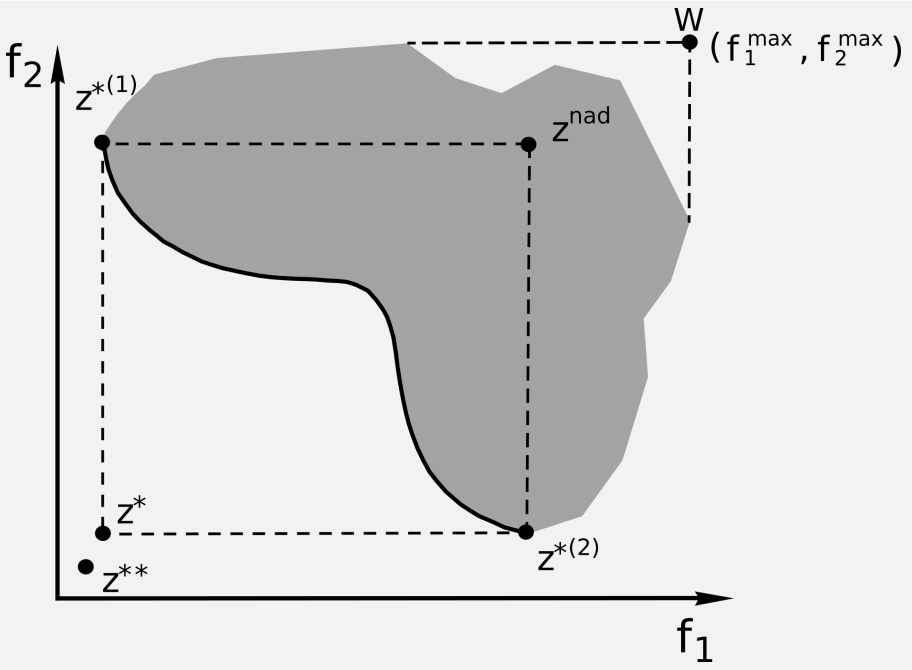

Figure 4.3. ***Special points for a MOOP****, namely the ideal* (z^*)*, utopian* (z^{**}) *and nadir* (z^{nad}) *objective vectors. Reprinted from [265].*

from the objective vector corresponding to the point having the worst feasible function values f_i^{max} in the entire search space. The latter, in fact, corresponds to the *worst point* "W" (see Figure 4.3). Furthermore, depending on the convexity and continuity of the Pareto-optimal set, the nadir point may represent a non-existent solution.

These special objective vectors are widely considered as reference points in many contexts, including *(i)* search algorithms, *(ii)* analytical methods, and *(iii)* performance metrics. For example, the knowledge of the nadir and ideal objective vectors is often used to normalise each objective function of the MOOP. In this way, it is possible to consider all the objective functions on the same footing. Indeed, if the ideal vector and nadir point of a MOOP are known, the classic expression for normalising the objective function is given by [267, 282]:

$$f_i^{norm} = \frac{f_i - z_i^*}{z_i^{nad} - z_i^*}. \tag{4.4}$$

A simple alternative for normalising the objective function values is to divide each objective

function by the corresponding value of the ideal point [267]. Notice that in the methods we shall present in the next sections, we will always consider objective functions that are properly normalised so that they can be regarded on the same footing.

4.3 Classical methods to solve MOOP

We refer to these methods, mainly to distinguish them from evolutionary methods, which we will discuss in Section 4.4. Methods for handling multi-objective optimisation problems can be classified according to different criteria. Several classifications exist in the literature [265, 267, 278, 281], the first one was provided by Cohon in Ref. [283]. Here, we refer to the fine-tuned classification presented in Ref. [267, 284] with the following four classes:

- No-preference methods;
- *A posteriori* methods;
- *A priori* methods;
- Interactive methods;

These names refer to the approach used to solve a MOOP. In mathematical terms, the problem is considered as solved when the Pareto-optimal set is found. Yet, in many practical contexts, our aim is to select only one particular solution in the non-dominated set. As a result, we must find a way to order the solutions in the Pareto-optimal set. Depending on the level of participation of the decision-maker, we can distinguish between different solution processes.

No-preference methods do not assume any information regarding the importance of each of the objectives, but a heuristic is required when selecting a single optimal solution. By contrast, *a posteriori* methods iteratively construct a set of Pareto-optimal solutions and preferential solutions are selected only at the end of the procedure. As the name suggests, *a priori methods*

rely on some initial preferences among the objectives and they usually lead to only one preferred Pareto-optimal solution. Last but not least, *interactive methods* progressively consider preference information during the optimisation procedure.

In the following, we only focus on describing some of the *no-preference* and *a posteriori* methods.

4.3.1 No-preference methods

When solving a MOOP using no-preference methods the information available to the decision-maker is neglected during the whole process. Hence, once the problem is solved by means of simple methods, the decision-maker can either reject or accept the solution found. As a result, these methods are often considered in contexts where the decision-maker merely accepts any optimal solution.

Let us start with the global criterion or compromise programming method [285]. The core idea is based on the minimisation of the distance between the feasible objective region and a special reference point.

Following the discussion presented in Ref. [267], we choose the ideal object vector as the reference point, and the L_p metric [286] for measuring distances. Thus, the MOOP translates to the *L_p-problem*, which is:

$$\begin{aligned} &\text{minimise} \quad \left(\sum_{i=1}^{k} |f_i(\mathbf{x}) - z_i^*|^p \right)^{1/p} \\ &\text{subject to} \quad \mathbf{x} \in S. \end{aligned} \tag{4.5}$$

The selection of the reference point is crucial in this context. In fact, this method cannot find better solutions than the reference point itself. It is therefore essential not to select a pessimistic reference point during the process. In particular, if the exponent $1 \leq p < \infty$, theoretical results show that the solution of the L_p-problem is Pareto- optimal [267]. Instead, if $p = \infty$, the problem

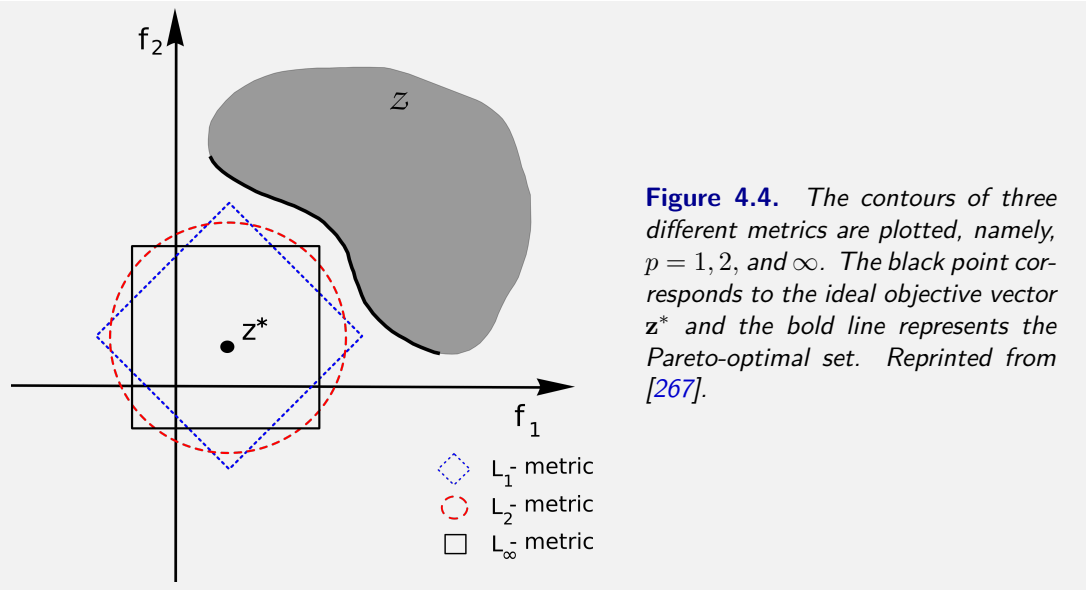

Figure 4.4. *The contours of three different metrics are plotted, namely, $p = 1, 2$, and ∞. The black point corresponds to the ideal objective vector $\mathbf{z}^*$ and the bold line represents the Pareto-optimal set. Reprinted from [267].*

is known as a *Chebyshev problem* and translates to:

$$\begin{aligned} &\text{minimise} \quad \max\,[|f_i(\mathbf{x}) - z_i^*|] \\ &\text{subject to} \quad \mathbf{x} \in S. \end{aligned} \tag{4.6}$$

Clearly, in the general case, the actual solution of the L_p-problem strongly depends on the metric selected. Common choices are for $p = 1, 2, \infty$. We report in Figure 4.4 an illustration of the global criterion for three different values of p.

It is worth remarking again that in the global criterion, the objective functions must be properly normalised. Otherwise, the objective functions that have the corresponding ideal objectives value closer to the feasible objective space will have a larger importance in the procedure. For instance, if the objective functions are bounded, they can be easily normalised as presented in Eq. (4.4).

4.3.2 A posteriori methods

A posteriori methods are often referred in the literature as *methods for generating Pareto-optimal solutions* [267]. The crucial steps of these methods can be grouped in two distinct phases. The first one, where the entire non-dominating set (or part of it) is iteratively generated. The generation process can be quite challenging and computationally expensive. The second part, instead, involves the decision-maker, who needs to select one of the solutions from a large set of possibilities. Here, we describe some of these *a posteriori* basic methods based on scalarisation, which are often considered as part of more complicated procedures.

The weighting method [265, 267, 278] represents one of the simplest scalarisation methods. The core idea entails converting the multiple objective functions of the MOOP into a single function by associating to each objective function f_i a weighting coefficient w_i. As a result, the goal translates to minimising the weighted sum of the objectives. Formally, we write it as [267]:

$$\begin{aligned} \text{minimise} \quad & \sum_{i=1}^{k} w_i f_i(\mathbf{x}) \\ \text{subject to} \quad & \mathbf{x} \in S \end{aligned} \tag{4.7}$$

where $w_i \geq 0$ for all $i = 1, 2, \ldots, k$ and $\sum_{i=1}^{k} w_i = 1$.

Yet, how to choose the values of the weights w_i? The answer is not unique. However, some theoretical results and simple considerations can be helpful. Analytical results [267, 281] guarantee that the solution found is Pareto-optimal if the weighting coefficients are strictly greater than zero or the solution is unique. Moreover, if the problem is convex, by fine-tuning the values of the weight w_i, it is possible to obtain all the Pareto-optimal solutions. By contrast, if the problem is non-convex, the weighting method does not allow us to retrieve all the Pareto-optimal solutions Notice again that in this method the objective functions f_i must be properly normalised, as in Eq. (4.4), for an adequate comparison between the objective functions.

To bypass the difficulties of the weighting method when solving problems with non-convex objective spaces, we may consider the ε- *constraints method* as an alternative. With this method, first introduced in Ref. [287], one of the objective functions is selected for optimisation, while all the others are considered as constraints of the problem with an upper bound for each of them. In other words, the MOOP transforms into [267]:

$$\begin{aligned} &\text{minimise} && f_l(\mathbf{x}) \\ &\text{subject to} && f_j(\mathbf{x}) \leq \varepsilon_j \quad \text{for all } \; j = 1, \ldots, k, \; j \neq l \\ & && \mathbf{x} \in S \end{aligned} \tag{4.8}$$

where $l \in \{1, \ldots, k\}$.

In principle, every Pareto-optimal solution can be found by means of the ε- constraint method simply by adjusting the upper bounds and the corresponding function to minimise. However, from a computational point of view, the method becomes far more time-consuming than the weighting methods. In addition, determining the appropriate bounds of the objective functions is non-trivial. To this end, the components of the ideal vector may become useful. For instance, by setting $\varepsilon_j = z_j^* + e_j$ for $j = 1, 2, \ldots, k, \; j \neq l$, where e_j represents a small non-negative real number that can be adjusted [267]. Finally, once the entire Pareto front is obtained, the decision-maker selects one (or multiple) points from the Pareto-optimal solutions by considering external qualitative functions. If external information is not available, a common choice consists in selecting solutions of the optimal set that have minimal distance from the ideal point [278]. More elaborate methods are based on (i) weighted combinations of distances using the ideal and nadir points as references [288, 289], (ii) clustering techniques [290], or (iii) ad-hoc normalised indexes (see [291] for a comparison).

Classical multi-objective optimisation algorithms are attractive since many theoretical results often support them. However, in practice, many of these methods may need to be used several

times with different tuning of the parameters to obtain different Pareto-optimal solutions. As a result, in practical applications, it is often preferred relying on other heuristics.

4.4 Evolutionary optimisation

Evolutionary algorithms (EAs) try to mimic evolutionary principles in the optimisation procedures. Unlike classical optimisation methodologies, the following main features characterise an EA [282, 292]:

1. Evolutionary optimisation (EO) procedures do not rely on gradient information in their exploration process,but are rather based on direct search procedures. As a result, they can be applied to a large variety of optimisation problems

2. Unlike most classical optimisation algorithms, an EO procedure mainly considers more than one feasible solution at a time for each iteration (i.e. a *population-based approach*).

3. Unlike deterministic operators adopted in classical optimisation techniques with fixed transition rules, *stochastic operators* represent the core engine of an EO procedure. In particular, these operators allow progressing towards different outcomes through biased probability distributions. As a result, EO algorithms construct a pool of feasible solutions with multiple and distinct optima, which usually allow them to maintain a global perspective in the search procedures.

The use of a population-based approach retains several advantages. For instance, it allows us to find and maintain distinct optimal solutions within the same iteration, which substantially helps to estimate the Pareto front of a MOOP correctly. More importantly, a population-based approach can be easily parallelised to speed-up the search procedures. However, a prominent disadvantage associated with a population of feasible solutions is the computational cost in terms of memory and operations for processing each iteration.

Here, we provide a brief overview of the crucial ingredients characterising evolutionary algorithms, following the discussion of Ref. [282]. Evolutionary optimisation algorithms start with a population of "individuals" that is usually created at random within certain bounds (if initially provided). After an initialisation procedure, the EO enters into an iterative core, usually composed of four principal operators [265, 282]: *evaluation*, *selection*, *variation* and *elitism*. After the evaluation of the current pool of candidate solutions, these operators are crucial for creating a new pool of individuals for each subsequent iteration. The algorithm halts when one or more termination criteria are satisfied. We report the pseudo-code of the entire procedure in Algorithm 1 [282].

Before entering into the details of the core operators, it is worth mentioning that any available knowledge regarding the problem can be effectively used when creating the initial population. As a matter of fact, custom pools are often useful to obtain fast convergence in search procedures [293]. The successive step of the evolutionary algorithm is the *evaluation* procedure. This includes a number of different micro-steps, corresponding to the computation of the values of each objective function and constraints, but also to assess whether the candidate solution is feasible or not. Moreover, since the evaluation of multiple functions could often be costly, some preference order can be establish when examining the pool of candidate solutions. For instance, when solving a MOOP, this ordering may correspond to the domination principle [292] based on some fitness functions, which usually reflect some combination of constraints, objective values, or any other internal attributes (e.g. concept of proximity among candidate solutions).

After evaluating all candidate solutions, the *selection* operator chooses some of them from the population according to some criteria. For example, in Ref. [294], the authors proposed the so-called *tournament selection*, which randomly selects two candidate solutions from the existing population and saves only the best for the successive iteration.

Subsequently, we enter in the *variation* procedure, which usually includes several stochastic

Algorithm 1: Evolutionary Optimization Procedure [282]

t = 0;
$Initialization(P_t)$;
do
  $Evaluation(P_t)$
  $P_t^{'} = Selection(P_t)$;
  $P_t^{''} = Variation(P_t^{'})$;
  $P_{t+1} = Elitism(P_t, P_t^{''})$;
  t = t+1;
while $(Termination(P_t, P_{t+1}))$;

operators, such as mutation and crossover, to produce a new pool of candidate solutions [265, 282, 292]. These operators are indeed the core component of the optimisation algorithm. A balanced fine-tuning of these operators is crucial for simultaneously exploring the decision space and for obtaining a quick convergence in the optimisation procedure. The *elitism* mixes all the candidate solutions (i.e. the existing and the newly created) and maintains only some of them.

A termination criterion determines when the evolutionary optimisation ends. A naive approach, for example, is based on a predetermined total number of iterations. In contrast, a more refined approach may take into account the variation of some statistical properties of the population over two (or more) consecutive iterations.

Among a vast literature on multi-objective algorithms [295–299], we mention the popular and widely accepted evolutionary algorithm, namely, NSGA-II [300] that will be used in Chapter 5.4.3 when introducing the Pareto growth model for multiplex networks.

4.4.1 Metrics of Pareto front performance

As the general outcome of multi-objective optimisation algorithms represents an approximation of the real Pareto-front, it becomes essential to quantitatively measure the performance of different algorithms. As a result, a large variety of performance indicators has been introduced over the last few decades [299, 301–303]. On the one hand, it is crucial to estimate the convergence to

the true Pareto front. On the other hand, instead, it is critical to measure several features of the approximated Pareto-optimal set. Examples of these features include the simple counts of the number of Pareto points, the distance from a reference set or measures of spread of the Pareto-optimal solutions in the objective space.

In the following, we shall present a measure introduced by Zitzler and Thiele in Ref. [304] that we will consider in Chapter 5.4. The *hypervolume indicator* (also known as Lebesgue measure [305] or S-metric [306]) is a popular quality measure for multi-objective optimisation. The function $\mathcal{S}$ is a measure of how much of the objective space is dominated by a given nondominated set $\mathbf{A}$.

Definition 4.4.1 (Size of the dominated space [306])

Let $\mathbf{A} = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_l) \subseteq \mathbf{X}$ *be a set of* l *decision vectors. The function* $\mathcal{S}(\mathbf{A})$ *[or equivalently* $I_H(\mathbf{A})$*] gives the volume enclosed by the union of the polytopes* $p_1, p_2, \ldots, p_l$*, where each* p_i *is formed by the intersections of the following hyperplanes arising out of* $\mathbf{x}_i$ *, along with the axes: for each axis in the objective space, there exists a hyperplane perpendicular to the axis and passing through the point* $(f_1(\mathbf{x}_i), f_2(\mathbf{x}_i), \ldots, f_k(\mathbf{x}_i))$*. In the two-dimensional case, each* p_i *is a rectangle defined by the points* (w_1, w_2) *and* $(f_1(\mathbf{x}_i), f_2(\mathbf{x}_i))$*, where* $\mathbf{W} = (w_1, w_2)$ *is the worst point.*

Notice that in principle, we might select other reference points when computing the hypervolume indicator, as for instance the nadir objective vector (see [307] for a discussion on the topic).

Nevertheless, the $\mathcal{S}$ measure has a major drawback. Given two Pareto Fronts obtained by two (possibly) different algorithms, the $\mathcal{S}$ measure cannot be used to assess whether either entirely dominates the other (see Fig. 4.5). As a consequence, the authors of Ref. [304, 306] introduced a second measure in order to properly compare two different Pareto fronts:

Definition 4.4.2 (Coverage of two sets [306])

Let $\mathbf{A}, \mathbf{B} \subseteq \mathbf{X}$ *be two sets of decision vectors. The function* $\mathcal{C}$ *maps the ordered pair* $(\mathbf{A}, \mathbf{B})$

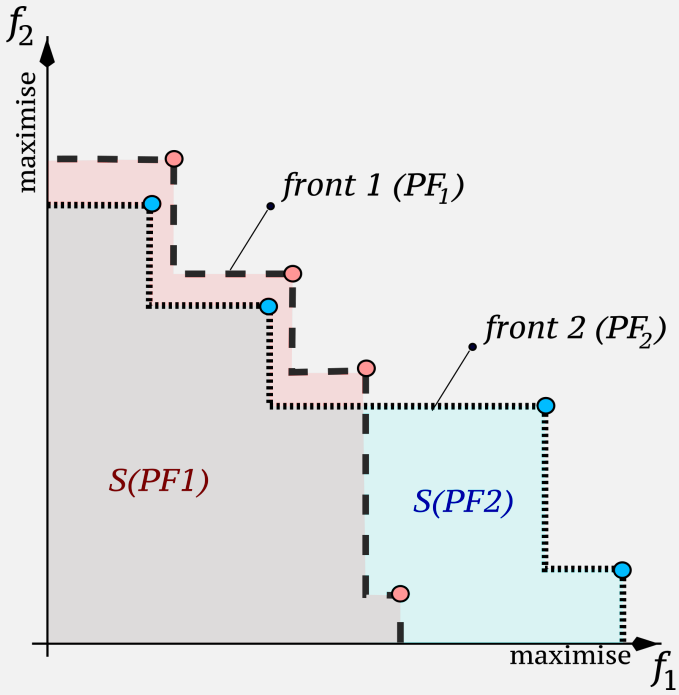

Figure 4.5. ***Metrics of performance for two different Pareto fronts***. *The hypervolume indicator $\mathcal{S}$ of the two fronts PF_1 and PF_2 is reported for a maximisation problem with two objective functions[a]. By analysing the coverage of the two sets, PF_1 is better than PF_2 since the former has only one point that is dominated by the other set. Adapted from [306].*

[a] In this maximisation problem, the point $(0, 0)$ is the worst point

to the interval $[0, 1]$*:*

$$\mathcal{C}(\mathbf{A}, \mathbf{B}) = \frac{|\{\mathbf{b} \in \mathbf{B} \mid \exists\, \mathbf{a} \in \mathbf{A} : \mathbf{a} \prec \mathbf{b}\}|}{|\mathbf{B}|} \tag{4.9}$$

where $\prec$ is the dominance relation introduced in Def. 4.2.5

Where, a value of $\mathcal{C}(\mathbf{A}, \mathbf{B}) = 1$ corresponds to the situation where all the decision vectors in $\mathbf{B}$ are dominated by $\mathbf{A}$. By contrast, if $\mathcal{C}(\mathbf{A}, \mathbf{B}) = 0$ then none of the solutions in $\mathbf{B}$ are dominated by $\mathbf{A}$. Both directions have to be considered, since in general $\mathcal{C}(\mathbf{A}, \mathbf{B}) \neq 1 - \mathcal{C}(\mathbf{B}, \mathbf{A})$.

Notice that in some cases the $\mathcal{C}$ measure might have problems in particular circumstances (i.e. the coverage for two sets could be equal but, at the same time, one of the two sets is actually closer to the Pareto-optimal front) that have been solved by the introduction of a new measure, namely the *coverage difference of two sets* [306].

Chapter 5

Models of multiplex networks

Models are essential guides for understanding the reality of systems and the processes that govern them [61]. On the one hand, null models provide a powerful tool to distinguish between what is and is not relevant in real-world systems. This is usually done by assessing whether structural features are under- (or over-) represented. On the other hand, models allow us to understand what are the main mechanisms responsible for the appearance of specific structural patterns.

In this chapter, we shall review several classes of models of multiplex networks. First, we focus on examining equilibrium models, namely, the canonical and microcanonical ensembles. We then move to another class of models, which aim at reproducing the structural patterns found in real-world multiplex systems.

Finally, in the last section, we present our main contribution, which is extensively based on our work [1]. We link multi-objective optimisation tools (i.e. Pareto principles) with multilayer network theory for the first time in the literature. In particular, we model the formation of multilayer transportation networks as a multi-objective optimization process, where service providers compete for passengers, and the creation of routes is determined by a multi-objective cost func-

tion encoding a trade-off between efficiency and competition. We then show that our model reproduces several structural features found in real-world systems as diverse as airline, train, and bus networks, thus suggesting that such systems are indeed compatible with the proposed local optimisation mechanisms.

5.1 Null models

5.1.1 Canonical and microcanonical ensembles

As mentioned in Section 2.1, tools coming from statistical mechanics are well suited for constructing models that both satisfy a set of constraints and are also the least possible biased. Given an empirical network, the main goal of this approach is to quantify how likely it is to observe a system with similar properties in an appropriately defined ensemble of random graphs whose elements satisfy a given set of constraints. Here, we provide a brief overview of the multiplex network ensembles following the discussion presented in Ref. [38, 212, 230]. Again, the microcanonical ensembles enforce a given set of hard constraints. By contrast, the canonical ensembles enforce a given set of soft constraints.

Let us consider a multiplex network $\mathcal{M}$ with M layers and N labelled nodes encoded in the vector of adjacency matrices $\mathbf{A}$. A multiplex network ensemble is defined by associating a probability $\mathrm{P}(\mathbf{A})$ to each of the possible configurations of multiplex network satisfying a given set of constraints. The Shannon entropy of the ensemble is defined as:

$$S = -\sum_{\mathbf{A}} P(\mathbf{A}) \log P(\mathbf{A}) \tag{5.1}$$

that is the logarithm of the typical number of multiplex systems in the ensemble.

In the case of independent layers, the probability $P(\mathcal{M}) \equiv P(\boldsymbol{A})$ of the multiplex network

can be factorised in the product of $P^{[\alpha]}(A^{[\alpha]})$ of each single layers, i.e.:

$$P(\boldsymbol{A}) = \prod_{\alpha=1}^{M} P^{[\alpha]}\left(A^{[\alpha]}\right). \tag{5.2}$$

Therefore, thanks to Eq. 1.9, the entropy of the multiplex ensemble can be written as:

$$S = \sum_{\alpha=1}^{M} S^{[\alpha]} = -\sum_{\alpha=1}^{M} P^{[\alpha]}\left(A^{[\alpha]}\right) \ln\left[P^{[\alpha]}\left(A^{[\alpha]}\right)\right]. \tag{5.3}$$

Let us now consider the case of the canonical multiplex ensemble - indicated by C - where soft constraints are imposed. In the case of multiplex networks, example of such constraints include the expected degree $k_i^{[\alpha]}$ of each node i in every layer α, the expected multidegree sequences $k_i^{\vec{m}}$ (see Chapter 2.2 for the formal definition), or the expected total number of edges on each layer $K^{[\alpha]}$, with $\alpha = 1, \ldots, M$. Let us consider R of these soft constraints such that:

$$\sum_{\mathbf{A}} P(\mathcal{M}) F_\mu(\mathcal{M}) = C_\mu \tag{5.4}$$

where $\mu = 1, \ldots, R$, and $F_\mu(\mathcal{M})$ is the function that describes how these constraints C_μ, with $\mu = 1, \ldots, R$, are imposed on the network.

As mentioned in Chapter 2.1, the least biased method of constructing multiplex networks with given structural properties is obtained by maximizing the entropy of the ensemble S, given the set of structural constraints [i.e. Eq. (5.4)]. As a consequence, by solving the optimisation problem, the probability $P_C(\mathcal{M})$ of observing the multiplex $\mathcal{M}$ reads:

$$P_C(\mathcal{M}) = \frac{1}{Z_C} \exp\left[-\sum_{\mu} \lambda_\mu F_\mu\left(\mathcal{M}\right)\right] \tag{5.5}$$

where the values of λ_μ represent the Lagrangian multipliers satisfying the constraints in Eq. (5.4), while Z_C is the partition function of the canonical multiplex ensemble. As a result, in the

canonical multiplex ensembles, the entropy can be written as:

$$S = \sum_{\mu} \lambda_{\mu} C_{\mu} + \ln Z_C. \tag{5.6}$$

By contrast, in the microcanonical ensemble - indicated as MC - the constraints are enforced exactly, so that the probability $P_{MC}(\mathcal{M})$ of the microcanonical multiplex ensemble can be expressed as:

$$P_{MC}(\mathcal{M}) = \frac{1}{Z_{MC}} \prod_{\mu=1}^{R} \delta_{F_{\mu}(\mathcal{M}),C_{\mu}} \tag{5.7}$$

where δ represents the Kronecker delta function and Z_M is the microcanonical partition function, which accounts for the number of multiplexes satisfying the hard constraints $F_{\mu}(\mathcal{M}) = C_{\mu}$, for $\mu = 1, \ldots, R$. The partition function is given by:

$$Z_{MC} = \sum_{\mathbf{A}} \prod_{\mu=1}^{R} \delta_{F_{\mu}(\mathcal{M}),C_{\mu}}. \tag{5.8}$$

We define the entropy of these multiplex ensembles as $N\Sigma$. In particular, by combining the definition of entropy in Eq. (5.1) and the expression of P_{MC} given in Eq. (5.7), it follows:

$$N\Sigma = -\sum_{\mathbf{A}} P_{MC}(\mathcal{M}) \ln P_{MC}(\mathcal{M}) = \ln Z_{MC} \tag{5.9}$$

where Σ is the Gibbs entropy of the multiplex ensemble. It has been shown in Ref. [230] that the Gibbs entropy Σ is related to the Shannon entropy S of the corresponding canonical ensemble (i.e. the conjugated canonical ensemble) by the following relationship:

$$N\Sigma = S - N\Omega \tag{5.10}$$

where Ω represents the logarithm of the probability that in the (conjugated) canonical multiplex ensemble the hard constraints $F_{\mu}(\mathcal{M})$ are satisfied.

Additionally, in Ref. [230] the author provides extensive analytical explanations on how to compute the entropy and partition function for a large variety of classes of multiplex networks with an increasingly stringent set of constraints in both canonical and microcanonical ensembles. These results include cases of independent and dependent layers (i.e. when Eq. (5.2) does not hold) to also account for multiplex systems with a prescribed amount of edge overlap. The same method has also been generalised to different multiplex structures, including multiplex systems with the heterogeneous activity of the nodes [308], and spatial multiplexes [309].

5.1.2 Randomisation algorithms

Along with the multiplex network ensembles discussed above, randomisation algorithms provide unique tools to construct null models starting from real-world data. In the context of multiplex networks, we can distinguish three main classes of randomisation procedures according to their aim. We follow the discussion considered in Ref. [38]:

1. The first class of algorithms aims at preserving the inter-layer degree correlations while simultaneously removing the effect of intralayer degree correlations and link overlap among layers. To this end, a double-edge swap procedure allows randomising each layer of the multiplex while leaving the degree sequence preserved. The core of the procedure unfolds in the following two steps:
   - Randomly select two edges of a random layer α and denote them as (i, j) and (k, l).
   - Swap the endpoints of these two edges [e.g. (i, k) and (j, l) or (i, l) and (j, k)] if and only if the swap does not create a multiple edge on layer α.
2. The second algorithm removes the effect of inter-layer degree correlation from a multiplex network. This is simply obtained by reshuffling the label of the nodes in each of the layers. In this way, the degree sequence of each layer is maintained while the existent inter-layer degree correlation is completely removed.

3. The third procedure allows maintaining the same amount of link overlap in a generic multiplex. To achieve this, we rely again on the double-edge swap procedure. The rewiring process is as follows:

   - Randomly select two edges of a random layer α with the same amount of link overlap and denote them as (i, j) and (k, l).
   - Swap the endpoints of these two edges [e.g. (i, k) and (j, l) or (i, l) and (j, k)] if and only if the swap does not create a multiple edge.

5.2 Models for reproducing structural properties

In this section, we provide an overview of a few models recently proposed in the literature for tuning and reproducing several structural features observed in real-world multiplex networks. For present purposes, we refrain from discussing a large class of models aimed at generating and reproducing mesoscale structures in multiplex networks [246, 252, 310, 311]

5.2.1 Rewiring model for tunable edge overlap

Starting from the double-edge swapping procedure it is possible to construct multiplex models that simultaneously preserve the degree distributions of each layer of the system and tune the amount of edge overlap. The model was first introduced in Ref. [41] for duplex systems and works along the following lines.

Let us start with two identical networks with K links and a maximum amount of edge overlap, i.e. $\omega = 1$. We obtain the desired value of edge overlap ω^* by iteratively rewiring the edges in only one of the two layers, while keeping the degree sequence fixed. During the procedure, we always select edges that appear in both layers. As a result, the number of edges with overlap equal to 2 decreases by one unit while the number of links present in only one layer increases by

two units. Hence, the overlap can be expressed in terms of the fraction r of the K edges rewired in one of the two layers as:

$$\omega = \frac{(1-r)K}{(1+r)K} = \frac{(1-r)}{(1+r)}. \tag{5.11}$$

As a consequence, by rewiring a fraction $r = (1-\omega)/(1+\omega)$ of edges in one of the two layers, it is possible to obtain a prescribed value of edge overlap ω. Note that with this model it is possible to obtain any value of overlap ω in the range $[0, 1]$, yet the inter-layer degree correlation ρ is always equal to one (i.e. maximally positive degree correlation).

As presented in Chapter 3.3 when discussing the complexity of a multiplex network, the same procedure can be easily generalised to the case of multiplex networks with an arbitrary number of layers. We have implemented for the first time this generalisation by iteratively selecting edges at random but ensuring that they have values of edge overlap greater than one [2]. Moreover, it is also possible to generalise the procedure described above in order to increase the edge overlap of a generic multiplex network with M layers. In this case, a rewiring is accepted only if it results in the increase of edge overlap of at least one of the two edges involved in the rewiring. As a result, this procedure also preserves the degree sequence of each layer.

5.2.2 Models of node and layer activity

As discussed in Chapter 2.2, both node and layer activity represent important structural features in multiplex networks. It is therefore reasonable to provide simple explanations to account for the heterogeneous distribution observed in real-world systems both for node and layer activities. Here, we review three basic models introduced in Ref. [224].

5.2.2.1 The hypergeometric model

In this model, we consider two layers α and β. We start with no active nodes in both layers and uniformly activate $N^{[\alpha]}$ nodes on layer α and $N^{[\beta]}$ nodes on layer β. Since the activity of a node at a given layer is uncorrelated from its activity at another layer, the probability of finding exactly m nodes active in both layers is given by the hypergeometric distribution:

$$P(m; N, N^{[\alpha]}, N^{[\beta]}) = \frac{\binom{N^{[\alpha]}}{m}\binom{N - N^{[\alpha]}}{N^{[\beta]} - m}}{\binom{N}{N^{[\beta]}}}. \tag{5.12}$$

Hence, the expected number of nodes active in both layers is equal to $N^{[\alpha]}\, N^{[\beta]}/N$, while the expected pairwise multiplexity (see Chapter 2.2 for the formal definition) can be expressed as:

$$\widetilde{Q}_{\alpha,\beta} = \frac{N^{[\alpha]} N^{[\beta]}}{N^2}. \tag{5.13}$$

By analogy, it is possible to quantify the expected value of the normalised Hamming distance as:

$$\widetilde{H}_{\alpha,\beta} = \frac{\sum_{m=0}^{N^{[\beta]}} \left(N^{[\alpha]} + N^{[\beta]} - 2m\right) \times p\left(m; N, N^{[\alpha]}, N^{[\beta]}\right)}{\min(N, N^{[\alpha]} + N^{[\beta]})}. \tag{5.14}$$

The empirical results presented in Ref. [224] show that this model does not produce an heterogeneous distribution of activities, which is instead present in many real-world multiplex systems.

5.2.2.2 Multi-activity deterministic and stochastic model

In this model [224], known as the multi-activity deterministic model, we create multiplex networks with the same number of active nodes N and layers M as those observed in an empirical system. In particular, a node i is considered to be active in the multiplex if it is active in at least one layer (i.e., $\mathbf{k}_i \neq \mathbf{0}$). We then associate to each node i of the multiplex a vector of node-activity, which

is uniformly sampled from all the possible $\binom{M}{B_i}$ M-dimensional vectors with exactly B_i non-null elements. As a result, we obtain multiplex networks that preserve the node activity distribution B_i of the empirical system, while destroying the effect of correlation in layer activity and the distribution of node-activity vectors. Clearly, since the vectors of node-activity are uniformly sampled, we know that all the layers will have the same expected number of active nodes:

$$\widetilde{N}^{[\alpha]} = \frac{1}{M}\sum_i B_i. \tag{5.15}$$

The authors of Ref. [224] also consider a slight variation of the previous model, known as the multi-activity stochastic model, where each node i at layer α is activated with probability $\overline{B}_i = B_i/M$. Also in this case, B_i represents the node-activity vector of node i observed in the empirical system. Even though the expected activity of each layer remains equal to $M^{-1}\sum_i B_i$, in this model the node-activity distribution is different from the one observed in the original network. This is because the node-activity of each node i is a binomial random variable distributed around B_i.

5.2.3 Relabelling model for tunable inter-layer degree correlation

The procedure for tuning the inter-layer degree correlation ρ was first presented in Refs. [224, 312]. Briefly, starting from a generic duplex network, we consider R to be the $N \times N$ matrix that accounts for the coupling between the nodes of the two layers. Here, the generic entry $r_{ij} = 1$ if node i in layer α corresponds to node j in layer β. Since we are dealing with a duplex network, there is a one-to-one correspondence between the nodes in the two layers, so that we have to impose $\sum_j r_{ij} = 1 \forall i$. The main idea is that the coupling R can be realised in many ways, and among all these possibilities we choose one that corresponds to a given level of inter-layer degree correlation $\rho^\star$. We define the cost function $F(R) = |\rho - \rho^\star|$, and we iteratively modify the structure of the matrix assignment in order to minimize $F(R)$. The minimisation procedure is

obtained by using a simulated annealing algorithm. In particular, starting from a certain matrix assignment R, we swap two elements of R at random in order to obtain a new assignment R'. We then accept the new assignment with probability:

$$p = \begin{cases} 1 & \text{if } F(R') < F(R) \\ \mathrm{e}^{-\beta\left[F(R')-F(R)\right]} & \text{otherwise} \end{cases}$$

where β has the role of an inverse temperature in the simulated annealing procedure. The algorithm stops when $F(R) < \varepsilon$, where ε is a threshold set by the user. We shall see this model once more in Chapter 6.3 when discussing the optimal percolation problem on correlated multiplex networks with overlap.

5.3 Growth models

Growing multiplex network models aim at providing simple dynamical explanations for the emergence of certain structural properties observed in real-world data sets. In the context of single-layer networks, for example, the Barabási-Albert (BA) model is one of the most famous at generating scale-free networks by means of a preferential attachment process [64]. Here, we shall discuss some of the models recently proposed in the literature, where simple dynamical mechanisms are able to produce layers with distinct types of degree distribution and are responsible for generating a different level of inter-layer degree correlations.

5.3.1 Linear and non-linear preferential attachment

One of the first multiplex growth models was proposed in Ref. [236], generalising the linear

preferential attachment mechanism to the case of multiplex networks. In particular, individual nodes join the network and choose to connect to older nodes j depending on a function of their degree k_j. However, in the multiplex network formalism, the degree of each node j is not a scalar, but a vector. As a result, the probability $\Pi_{i\to j}^{[\alpha]}$ that a new node i attaches to a node j depends on all the degree components in the layers, i.e. [212]:

$$\Pi_{i\to j}^{[\alpha]} = \frac{F_j^{[\alpha]}(\mathbf{k_j})}{\sum_l F_l^{[\alpha]}(\mathbf{k_l})}. \tag{5.16}$$

In Ref. [236], the authors presented the analytical treatment for a growing multiplex network model with two layers and a linear attachment kernel. This corresponds to setting $F_j^{[\alpha]}$ as a convex combination of the degrees of node j at all the layers. Interestingly, this model produces multiplex networks where the two layers have scale-free degree distributions (with an exponent $\gamma = 3$ in the thermodynamic limit), and the system displays positive inter-layer degree correlations among the layers. This is because the expected final degree of a node on certain layers solely depends on the time at which it joined the network [236]. The generalisation of this model to the case of multiplex networks with more than two layers is discussed in Ref. [313].

A more refined model was proposed in Ref. [314] by considering a non-linear attachment kernel [315]. Based on that model, it is possible to generate layers with different degree distributions and to produce negative inter-layer degree correlation. Also in this case, each new node is simultaneously added to both the layers of a duplex network. Each new node is connected to the other nodes of the same layer by m links. The probability $\Pi_{i\to j}^{[\alpha]}$ that a new node i attaches to j on a given layer α can be expressed as:

$$\Pi_{i\to j}^{[\alpha]} \propto \left(k_j^{[1]}\right)^{r_1} \left(k_j^{[2]}\right)^{r_2} \tag{5.17}$$

where $r_1, r_2 \in \mathbb{R}$, $\alpha \in \{1, 2\}$. As observed in the case of single-layer networks [315], the non-linear attachment mechanism generates a rich variety of behaviours. For instance, by varying

the exponents r_1, r_2, it is possible to generate systems with negative, null and positive inter-layer degree correlation. While the layers may display either a power-law, exponential, or a condensed degree distribution. The authors also propose possible generalisations of the model for systems with more than two layers.

5.3.2 Layer-based growth models

Another class of growth models concern those in which new layers sequentially join into the system. Their main aim is to explain the fat-tail distributions of node and layer activity observed in real-world systems [224], but also the scale-free degree distribution on the aggregate networks. Here, we review two classes of models introduced in Refs. [224, 226].

The first example of modelling multilayer networks was provided by Criado et al in 2012 [226]. Rather than providing simple mechanisms for reproducing structural patterns observed in real-world multilayer networks, the authors investigated the importance of the whole multilayer structure compared to the corresponding single-layer aggregate. More precisely, they proposed several stochastic mechanisms for constructing multilayer systems with distinct mesoscale structures and show that these systems often lead to aggregate networks with similar characteristics. As a result, when analysing a multilayer system, it is imperative to take into account the mesoscale structure of the entire structure rather than focusing on the aggregate representation.

The second model was introduced in Ref. [224] and works as follows. At time $t_0 = 0$, we consider a multiplex with M_0 layers and N nodes. Then, at each time steps t, a new layer α with $N^{[\alpha]}$ nodes to be activated joins the system. Here, the quantity $N^{[\alpha]}$ can be taken from the data set we would like to reproduce. In particular, at time t each node i has a probability of being active on that layer equal to:

$$p_i(t) = B_i(t) + A \tag{5.18}$$

where $B_i(t)$ represents the number of layers where the node i is active in the multiplex (i.e. the node activity at time t), while $A \in \mathbb{R}$ is a non-negative constant that allows a node to be activated with a non-null probability (i.e. to activate nodes not yet active in the system). Remarkably, when the number of layers in the system increases, the layer activity $P(N^{[\alpha]})$ becomes closer to a power-law like distribution.

5.4 Pareto growth model

As discussed in the previous sections, models represent a powerful tool to characterise and investigate a certain set of structural features in real-world systems. Here, we present our main contribution [1] for the modelling of real-world transportation multiplex networks, by combining tools from multi-objective optimisation and multilayer network theory. In particular, we highlight that this is the first growth model in the network literature, which links multi-objective optimisation theory with the multilayer network representation. In particular, transportation systems can be represented as multiplex networks, where each layer represents the routes served by a single service provider. Hence, the core idea of our model is inspired by the observation that the formation of edges in many real-world transportation networks [14, 316] is often subject to concurrent spatial and economical constraints [317–319]. On the one hand, there is the tendency to accumulate edges around nodes that are already well-connected, in order to exploit the economy of scale associated to hubs. On the other hand, each service provider usually tends to minimise the competition with other existing service providers. As we shall discuss through the section, these two mechanisms provide a reasonable explanation for the emergence of highly-optimized heterogeneous multiplex networks. In addition, this represents the first example where Pareto efficiency is extensively used for modelling in multilayer network theory.

Let us consider a multiplex transportation network with N nodes and M layers, where nodes represent locations and layers represent service providers, e.g. airline, train or bus companies.

Each layer is the graph of routes operated by one of the service providers. Using the multiplex formalism introduced in Chapter 2.2, the network can be described by a set of adjacency matrices $\{A^{[1]}, A^{[2]}, \dots, A^{[M]}\} \in \mathbb{R}^{N\times N\times M}$, where the entry $a_{ij}^{[\tau]}$ is equal to 1 if i and j are connected by a link at layer τ (meaning that provider τ, with $\tau = 1, 2, \dots M$ operates a route between location i and j), while $a_{ij}^{[\tau]} = 0$ otherwise[1]. Here, $k_i^{[\tau]}$ represents the degree of a node i at the layer α, $K^{[\tau]}$ is the total number of links of layer τ, o_i is the total degree of node i, and o_{ij} represents the overlap of edge (i, j) [216, 230].

For this model, we assume that service providers join the system one after the other, each one with a predetermined number of routes that they can operate. This means that the multilayer network acquires a new layer at each (discrete) time step τ. When the layer joins the system, the new provider tries to place its routes in order to maximise its profit. To this end, a provider would prefer to have access to as many potential customers as possible (i.e., to connect locations with large population), whilst minimising the competition with other providers (i.e., to avoid to operate a route if it is already operated by other providers). In order to mimic these two competing mechanisms, we set the probability of creating an edge between node i and node j at the new layer τ as:

$$p_{ij}^{[\tau]} \propto \frac{o_i^{[\tau-1]} o_j^{[\tau-1]} + c_1}{o_{ij}^{[\tau-1]} + c_2} \qquad \tau = 2, \dots, M \tag{5.19}$$

where $o_i^{[\tau-1]}$ and $o_{ij}^{[\tau-1]}$ are respectively the overlapping degree of node i and the edge overlap of (i, j) at the time step $\tau - 1$. The non-negative constants c_1 and c_2 allow a non-zero probability of creating a new edge to a node that is isolated in all the existing layers. The rationale behind Eq. (5.19) is that the total degree o_i of node i can be used as a proxy of the population living at that location. Hence, in the same spirit of the "gravity model" [320, 321], creating a link between node i and node j with a probability proportional to the product $o_i\, o_j$ will increase the chances of a provider gaining access to a large set of customers. Similarly, by requiring that $p_{ij}^{[\tau]}$ is inversely

[1]Notice that throughout the section we are going to use τ to denote the generic layer (instead of α, as done in the previous chapters), since it will simultaneously represent the layer and the timestep in the growth model. Such choice is also consistent with the original notation introduced in the original paper [1].

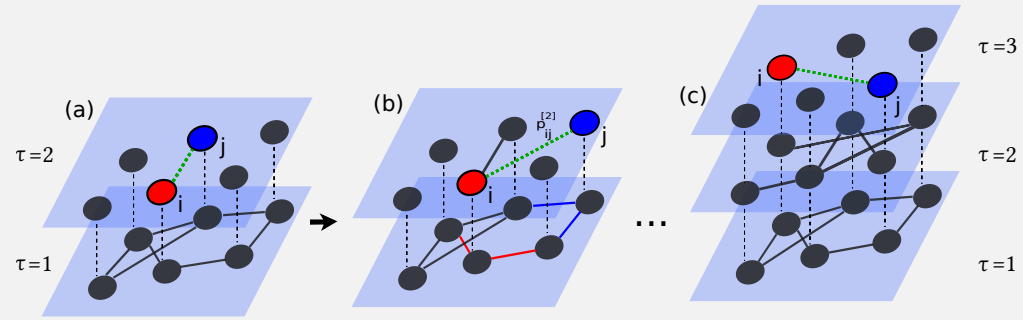

Figure 5.1. ***Schematic illustration of the MMOO growth model.*** *(a) At the step $\tau = 2$, a new layer arrives with $K^{[2]}$ links to be placed, and the first edge is placed uniformly at random among all possible pairs of nodes. (b) The remaining $K^{[2]} - 1$ links are placed according to the probability $p_{ij}^{[2]}$ in Eq. (5.19). (c) The same procedure is repeated for each layer τ until a multiplex with M layers is obtained.*

proportional to the edge overlap $o_{ij}^{[\tau-1]}$ we discourage the creation of a new route between two locations if they are already served by a large number of other providers, thus modelling the tendency of providers to avoid competition. The two competing mechanisms we propose can be formalised as a MOOP:

$$\begin{cases} \max \mathbf{F} \\ \min \mathbf{G} \end{cases} = \begin{cases} F^{[\tau]} = \displaystyle\sum_{i,j:\, a_{ij}^{[\tau]}=1} (o_i\, o_j + c_1) \\ G^{[\tau]} = \displaystyle\sum_{i,j:\, a_{ij}^{[\tau]}=1} (o_{ij} + c_2) \end{cases} \tag{5.20}$$

where the efficiency function $F^{[\tau]}$ accounts for the number of potential customers, while $G^{[\tau]}$ measures the competition due to route overlaps. Given the parallelism with multi-objective optimisation, we will refer to this model as the Multiplex Multi-Objective Optimisation (MMOO) model.

A schematic illustration of the model is reported in Fig. 5.1. The first layer is a connected random graph with $K^{[1]}$ edges. At each step τ, with $\tau = 2, \ldots, M$, a new layer is created. The first of the $K^{[\tau]}$ edges of the new layer is placed uniformly at random among the $\binom{N}{2}$ possible

edges. In order to obtain a connected network, the remaining $K^{[\tau]}-1$ links are created according to the probability in Eq. (5.19), yet ensuring that one of the two endpoints of the selected edge belongs to the connected component at that layer. The total number of links at each of the M layers are external parameters of the model. Considering the routes of each provider as fixed over time may seem an unrealistic oversimplification, but in all systems considered, providers normally update their network routes at a slow rate. This coheres with our assumption that the routes of each layer are quasi-static. In fact, rearranging a set of train services or flights entails substantial logistic and economic investments, since railway licenses and airport-slots are normally allocated for long periods of time (for several years).

Before presenting the empirical results of our model, let us introduce the three different data sets considered. Notice that two of them, namely, the UK trains and UK coaches data sets, were first used in our study [1] and are available in Ref. [322].

5.4.1 Transportation network data sets

The multiplex transportation systems considered for this study are: (i) the undirected routes of the six continental airlines (OpenFlight), (ii) the UK national railway network, and (iii) the UK national coach network. These data sets represent the main modes of public transportation networks. We describe below these data sets and summarise in Table 5.1 the basic structural properties of the corresponding multiplex networks. Additional details for the continental multiplex airlines can be found in Ref. [224], while the data sets are available at [322, 323]. We considered all these multiplexes as unweighted.

5.4.1.1 OpenFlight

The networks of aerial routes originally used in Ref. [224] were constructed from the collaborative free online tool OpenFlight [324], which allows one to map flights all around the world. For each

Multiplex	N	M	$\langle N^{[\alpha]} \rangle$
Africa	238	84	9.8
Asia	795	213	24.4
Europe	593	175	21.8
North America	1029	143	24.9
Oceania	261	37	14.1
South America	300	58	15.1
UK Coaches	11738	1207	16.6
UK Trains	1658	41	64.27

Table 5.1. ***Basic structural properties of the multiplex transportation data sets***. *The number of nodes N, number of layers M and the average layer activity $\langle N^{[\alpha]} \rangle$ are reported for the eight multiplexes analysed.*

route, the dataset contains information about the starting airport, the destination airport, and the company operating each flight. Six different multiplex networks were constructed, one for each continent (Africa, Asia, Europe, North America, Oceania, South America), each consisting of as many layers as airlines operating in that continent. The edges on each layer are the direct routes operated by the corresponding airline, and the active nodes on each layer are the airports that are connected by at least one route operated by that company.

5.4.1.2 UK trains

We considered the timetables extracted from OpenTrainTimes [325], which contain real-time information about the routes operated by 41 different railway companies over 2944 stations across the UK. In this case, nodes represent railway stations, while a link exists between two nodes only if there is at least one service between them. The routes of OpenTrainTimes are regularly updated using the real-time data feeds system provided by the official network rail website [326]. For each route, we have information about the starting station, the end station, the company which operates the service, and also the days in which they operate their routes. For the analysis done in the main text, we have constructed a multiplex network with 1658 nodes (the stations were aggregated at the city level) and 41 layers representing the different companies. It is worth mentioning that the UK train multiplex network mainly includes the 41 (regional) railway

companies operating in the UK. As a result, the service providers will only have a moderate overlap over the routes (as we will see later, the maximum overlap is 8). Notice also that the competition in terms of route formation for the UK train multiplex can be considered only from the service-provider point of view. By contrast, the competition in the UK train multiplex has no meaning from an user perspective.

5.4.1.3 UK coaches

We used the UK coach data set available at Ref. [327], which includes data from the National Coach Services Database (NCSD). The dataset contains information about 17433 routes between 12767 locations operated by 1219 different regional operators. In this case, nodes represent coach stations or stops, while a link exists between two nodes only if there is at least one connection between them. Layers represent the different regional service providers. Notice that for each route we have information about the start, middle and end points, as well as the regional operator code. Starting from this dataset we constructed a multiplex network having 1207 regional service providers (layers) operating over 11738 stops (nodes). The analysis does not include providers operating circular services.

5.4.2 Empirical results

We have used the MMOO model to reproduce the structure of the three multiplex transportation systems mentioned above. For each network, we generated 10^3 independent permutations of the sequence $\{K^{[1]}, K^{[2]}, \ldots, K^{[M]}\}$ of the total number of links at each layer in the data set. Then, for each permutation, we ran 50 independent realizations of the model. In our simulations we used a Metropolis-Hastings algorithm [328] to sample form the distribution in Eq. (5.19). We have also considered 10^3 realisations of a baseline model, i.e. an ER multiplex model, such that each layer α is generated as an ER random graph with N nodes and $K^{[\alpha]}$ links. For each

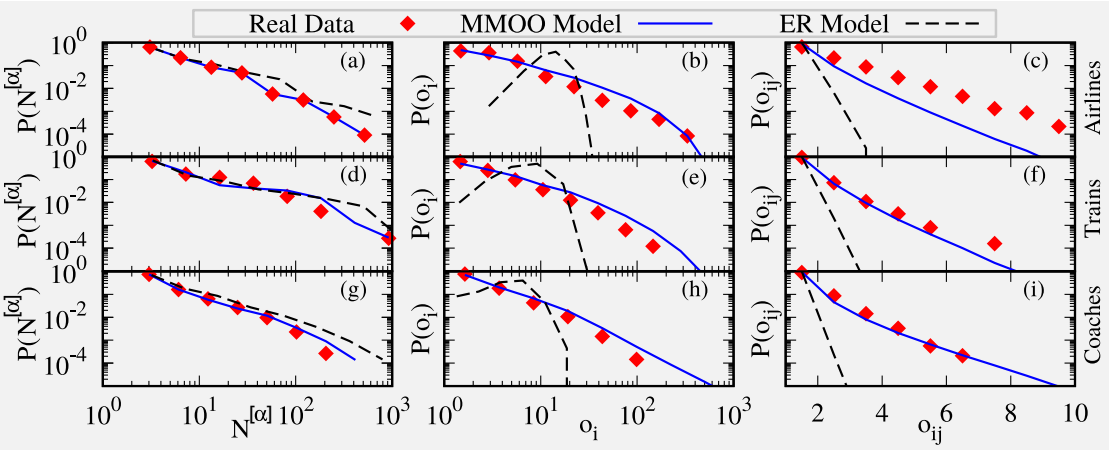

Figure 5.2. ***Performance of the MMOO model in reproducing the structural measures of three different transportation systems.*** *Distributions of layer activity $N^{[\alpha]}$ (left column), node total degree o_i (middle column) and edge overlap o_{ij} (right column). The multiplex networks (red diamonds) of North America airlines (top row), UK train services (middle row) and UK coach services are compared to the corresponding multiplex networks generated by the MMOO model (solid blue lines) and to multiplex networks whose layers are Erdös-Rényi graphs [62] (dashed lines). The results shown are averaged over 10^3 realizations (standard deviations are indistinguishable from the symbols).*

realisation, we consider an independent permutation of the sequence $\{K^{[1]}, K^{[2]}, \ldots, K^{[M]}\}$ of the total number of links at each layer in the data set. Notice that other models generating scale-free or Watts-Strogatz networks (or even mixture of them) on each layer of the multiplex might be better suited for comparisons. However, we leave this analysis for future works.

In Fig. 5.2, we report the distributions of layer activity $N^{[\alpha]}$, total node degree o_i, and edge overlap o_{ij} of the multiplex networks obtained with the MMOO model, where we set $c_1 = c_2 = 1$. The two-sample Cramer-von-Mises statistical test [329] provides convincing evidence that the synthetic distributions are compatible with the original ones (p-value < 0.01, except for panel (c), where $p < 0.5$). It is worth noticing that the MMOO model naturally reproduces the heterogeneous distribution of node total degree and the decreasing exponential behaviour of the edge overlap o_{ij}, which respectively mirror the heavy-tailed distribution of city size [330, 331] and the tendency of service providers to reduce the competition on single routes [56].

To extensively compare the synthetic and empirical distributions for the eight transportation

systems, we also investigate two other structural measures, i.e., the node activity B_i and pairwise inter-layer correlation $H_{\alpha,\beta}$ (see Chapter 2.2 for a formal definition). We report the p-values of the corresponding two-sample Cramer-von-Mises test in Table 5.2. For most of the multiplex analysed, the MMOO model with $c_1 = c_2 = 1$ performs quite well for at least two structural metrics. In particular, we find that the synthetic and empirical distributions of layer activity $N^{[\alpha]}$ and node total degree o_i are indistinguishable in almost all the data sets. Yet, it is evident that the MMOO model cannot reproduce all the structural measures of the different real-world systems, as confirmed by the high p-values found for B_i and $H_{\alpha,\beta}$.

In principle, it could be possible to fine-tune the values of c_1 and c_2 in Eq. (5.19) to accurately reproduce the structural properties of the multiplex transportation networks. To test the dependence of c_1 and c_2 on the model, we performed extensive Monte Carlo simulations where the two constants c_1, c_2 were tuned in the range $(0, 10)$ with steps of 0.1. In Table 5.3 we report, for each system, the values of c_1 and c_2 for which the two-sample Cramer-von-Mises test provides the best (lowest) p-values for the distributions of the five different structural metrics considered. In other words, we select the values of c_1 and c_2 which provide the lowest Euclidean distance between the five-dimensional point having as components the p-values (of the two-sample Cramer-von-Mises test) of the five different structural metrics and the ideal point $z^* = (0.001, 0.001, 0.001, 0.001, 0.001)$.

5.4.3 Pareto efficient providers and system efficiency

By considering the multi-objective optimisation framework formally defined in Eq. (5.20), it is possible to compare providers by looking at their position in the efficiency-competition plane defined by the two functions F and G and shown in Fig. 5.3. We focused on the air transportation networks and extracted from the empirical data the *observed Pareto front* of each continental network, i.e. the set of all the non-dominated points in the F-G plane. Surprisingly, we found that

Multiplex	o_i	o_{ij}	$N^{[\alpha]}$	B_i	$H_{\alpha,\beta}$
Africa	< 0.01	0.15	< 0.01	0.2	0.2
Asia	< 0.01	0.5	< 0.01	0.05	0.05
Europe	< 0.01	< 0.01	< 0.01	0.05	0.025
North America	< 0.01	0.5	< 0.01	0.25	0.75
Oceania	0.1	0.15	< 0.01	0.75	< 0.01
South America	< 0.01	0.75	< 0.01	0.1	0.25
UK Coaches	< 0.01	< 0.01	< 0.01	0.75	0.75
UK Trains	< 0.01	< 0.01	< 0.01	0.5	0.05

Table 5.2. ***Performance of the MMOO model in reproducing multiplex structural measures.*** *We report the list of the p-values obtained for the two-sample Cramer-von-Mises test when comparing the synthetic and empirical distributions of the eight transportation systems analysed. Five different structural measures were tested, namely, the node overlap (o_i), the edge overlap (o_{ij}), the layer activity ($N^{[\alpha]}$), node activity (B_i) and the normalised Hamming distance ($H_{\alpha,\beta}$). Notice that we used $c_1 = 1$, $c_2 = 1$ in the MMOO model for obtaining the synthetic distributions.*

Multiplex	o_i	o_{ij}	$N^{[\alpha]}$	B_i	$H_{\alpha,\beta}$	c_1	c_2
Africa	< 0.01	< 0.01	< 0.01	0.1	0.25	0.2	9.3
Asia	< 0.01	< 0.01	< 0.01	< 0.01	0.1	0.2	3.0
Europe	< 0.01	< 0.01	< 0.01	< 0.01	0.025	1.0	8.2
North America	< 0.01	< 0.01	< 0.01	0.1	0.75	0.1	9.0
Oceania	0.025	0.05	< 0.01	0.75	< 0.01	1.8	6.0
South America	< 0.01	< 0.01	< 0.01	0.1	0.5	0.2	9.0
UK Coaches	< 0.01	< 0.01	< 0.01	0.5	0.75	4.0	6.5
UK Trains	< 0.01	< 0.01	< 0.01	0.25	< 0.01	2.1	5.3

Table 5.3. ***Best values of c_1 and c_2 for the MMOO model when reproducing the empirical structural metrics.*** *Values of c_1 and c_2 for which we obtain the best (lowest) p-values of the Cramer-von-Mises test for all the five structural multiplex metrics.*

most of the Pareto-optimal points correspond to the most important companies in the continent (e.g., flagship, mainline, and large low-cost carriers), and in particular with those carrying the largest number of passengers.

In order to quantify the potential improvement attainable by a system in the F-G plane, we used NSGA-II, an evolutionary multi-objective optimization algorithm [300], to generate 10^5 synthetic multiplex networks for each continent. We then computed the *theoretical Pareto front*, consisting of the Pareto-optimal points resulting from all the simulations, reported in Fig. 5.3 as a dashed blue line. The closer the observed PF is to the theoretical PF, the better the system approaches the best possible solution in the F-G plane. Interestingly, the observed PF of the

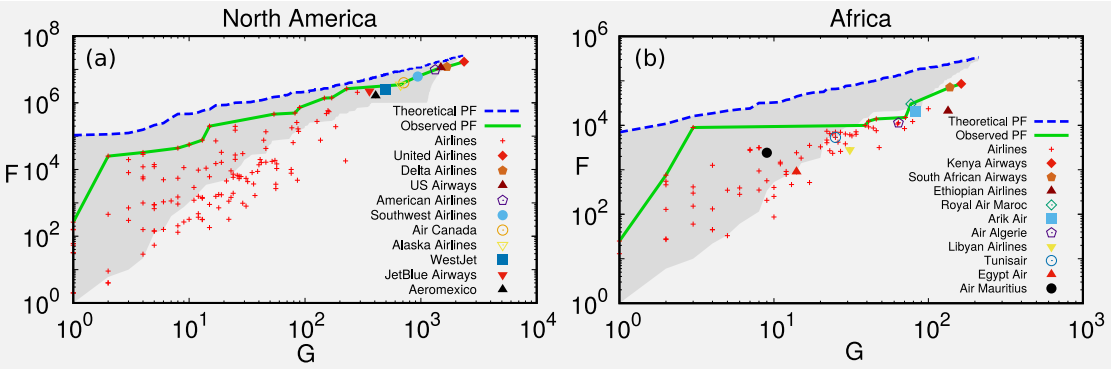

Figure 5.3. ***Pareto optimal companies in the F-G plane for two multiplex airline systems.*** *We report the observed (solid green line) and theoretical (dashed blue line) Pareto fronts for the continental airline networks of (a) North America and (b) Africa. The top ten airlines by number of passengers in 2013 are highlighted. For each system, the theoretical Pareto front was obtained as the non-dominated points of 10^5 realisations of the MMOO model (the range of variability of the simulations is indicated by the shaded grey region).*

North American airline network is relatively closer to its theoretical PF, while for the African airlines we observe a larger gap between the two curves. This means that, on average, the African airlines may obtain a greater improvement in the F-G plane than North American companies. As discussed in Chapter 4.4.1, a quantitative way to associate a number to a Pareto Front $\mathcal{P}$ is by means of the S-metric or hypervolume indicator $I_H(\mathcal{P})$. In a two-dimensional plane this quantity corresponds to the Lebesgue measure of the union of the rectangles defined by each point in the front $\mathcal{P}$ and a reference point [306, 332]. In our computation, we considered the worst point

Continent	Δ_H	GDP (US$) [333]	M	Pareto points
North America	$6.82 \cdot 10^{-5}$	17892	143	20
Asia	$7.79 \cdot 10^{-5}$	14070	213	25
Europe	$1.07 \cdot 10^{-4}$	36784	175	26
South America	$5.50 \cdot 10^{-4}$	9359	58	15
Oceania	$5.81 \cdot 10^{-4}$	13064	37	11
Africa	$5.82 \cdot 10^{-4}$	2843	84	10

Table 5.4. ***Hypervolume and economical performance for the continental airline networks.*** *The potential improvement attainable by the North American transportation system in the $F - G$ plane is measured by the normalised relative hypervolume Δ_H, where smaller values of Δ_H correspond to more optimized networks. We report the ranking of continents by Δ_H along with the economical performance of each continent (as measured by the average GPD per capita), the number of airlines M, and the total number of Pareto points. We also report the Kendall's correlation between the normalised hypervolume Δ_H and the number of airlines M ($\tau_b = -0.41, p \approx 0.25$), and with the number of Pareto points ($\tau_b = -0.6, p \approx 0.14$), respectively.*

as the reference point [in our case, this is the point of coordinates ($G_{\max}$, $F_{\min}$)]. The distance between an observed PF, $\mathcal{P}^{\text{obs}}$, and the corresponding theoretical PF, $\mathcal{P}^{\text{th}}$, can be quantified through the relative normalised hypervolume $\Delta_H = |I_{H_{\mathcal{P}^{\text{obs}}}} - I_{H_{\mathcal{P}^{\text{th}}}}|/(I_{H_{\mathcal{P}^{\text{th}}}} K)$. By dividing the relative hypervolume difference by the total number of routes K, it is possible to compare the level of potential improvement of two multiplex networks with respect to their corresponding theoretical PF. We argue that the value of Δ_H can be used to determine the level of optimality of a transportation system, with smaller values of Δ_H indicating more optimised configurations. It is important to highlight that since Δ_H represents the normalised relative hypervolume, the choice of the reference point when computing I_H has a minimal impact on its actual value.

Table 5.4 reports the ranking of continents induced by Δ_H, where North America and Asia

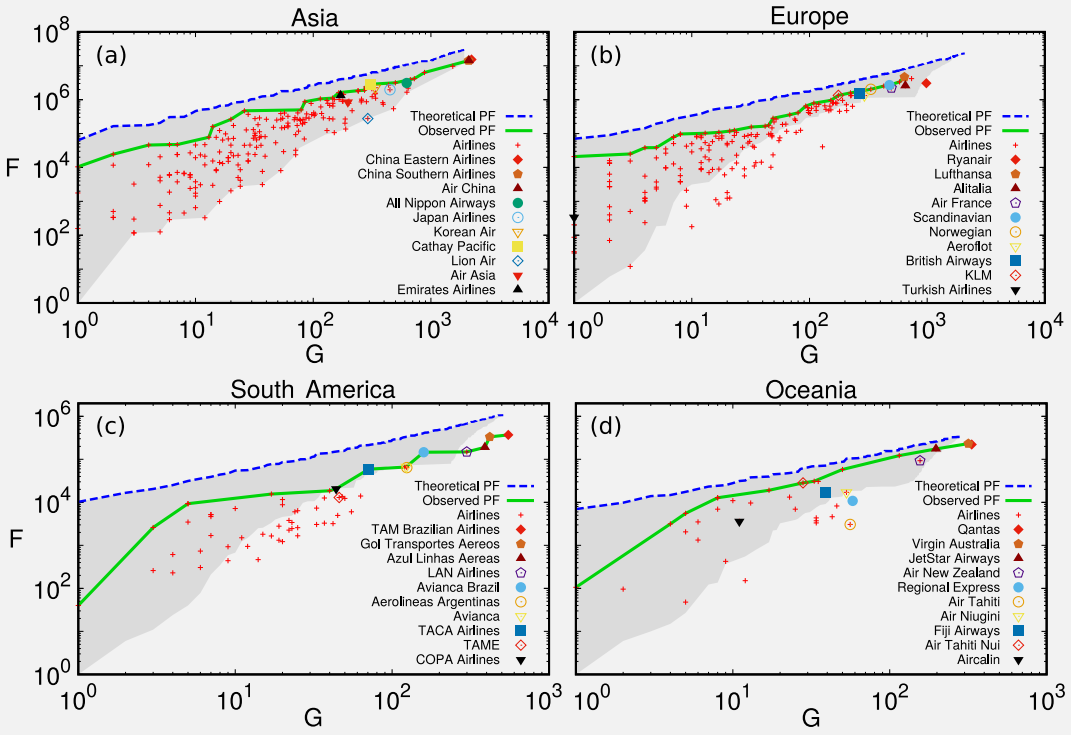

Figure 5.4. ***Performance of the continental airlines networks in the*** F-G ***plane.*** *The observed Pareto front (solid green line) along with the top ten airlines by number of passengers for Asia (a), Europe (b), South America (c), and Oceania (d). For each continent, the Theoretical Pareto front (dashed blue line) is obtained by considering the non-dominated points of* 10^5 *realisations of the MMOO model (the range of variability obtained through the simulations is indicated by the shaded grey region). Notice that in almost all the cases, the top airlines of a continent belong to the observed Pareto front.*

lead the pack, while Africa is lagging behind. Interestingly, that ranking is correlated with the ranking induced by continental GDP per capita of 2013 [333] (Kendall's $\tau_b = -0.6$, $p \approx 9 \cdot 10^{-2}$). However, the same result holds when considering economical data from the period 2010-2014. Interestingly, while a lower correlation appears between the number of airlines in the system and the corresponding Δ_H (Kendall's $\tau_b = -0.41, p \approx 0.25$), we found a moderate correlation between the number of Pareto points and Δ_H (Kendall's $\tau_b = -0.6, p \approx 0.14$), i.e. the higher the number of Pareto points the lower the values of ΔH. Yet, in these two cases, notice that the p-values are greater than 0.1.

For the sake of completeness, we report in Fig. 5.4 the results of the other four continental airline systems in the F-G plane. Also in this case, the top ten airlines by number of passengers belong to the observed Pareto front obtained from real data.

5.4.3.1 Pareto efficient strategies

Here, we show that our model can in principle be used by new companies entering the market, as a guide to place their routes in the most effective way. In particular, we can slightly modify the model as follows. We may consider all the layers as fixed and identical to the observed ones except for one of them, which represents a new service provider. We can then place all the edges of this provider (i.e. as the last layer of the system) according to Eq. (5.19).

We ran our model for several carriers and we report in Fig. 5.5 the results of our simulations. Remarkably, we found that the companies appearing on the observed Pareto front cannot improve their position in the F-G plane, meaning that their routes have evolved over time according to an effective optimization process. By contrast, our model is able to improve the position in the F-G plane of sub-optimal and non-optimal airlines.

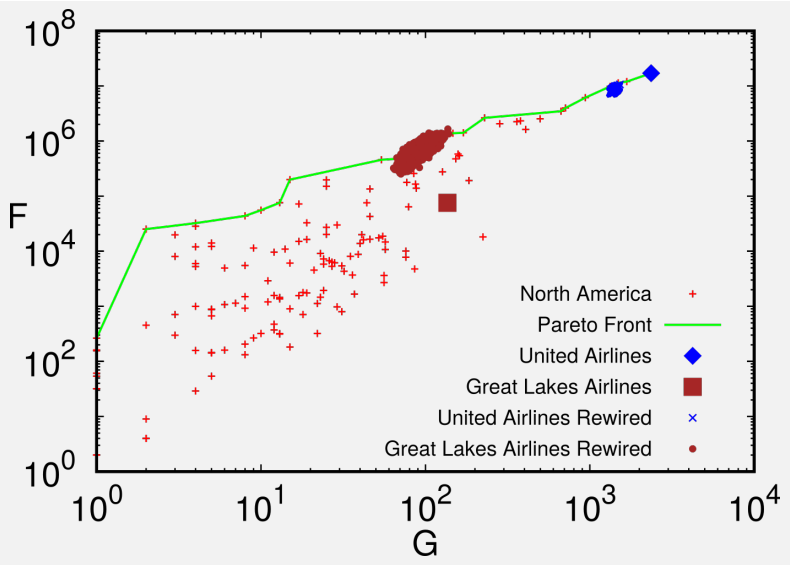

Figure 5.5. ***Pareto efficient strategies for existing carriers.*** *We reshape the routes of two North American airlines in the F-G plane. United Airlines, a Pareto-optimal company (blue diamonds), cannot be improved any further by our model, while Great Lakes Airlines (brown squares) can potentially get much closer to the Pareto front.*

Chapter 6

Optimal percolation in correlated multiplex networks with overlap

The discovery of interdependencies and feedback loops between interacting networks have completely shaken percolation theory. While in single-layer percolation theory the failure of a node only affects its neighbourhood, in multiplex networks the failure of nodes in one network may lead to the failure of their replicas (i.e. the replica nodes) in the other layers. As a consequence, in presence of interdependencies, multiplex networks are in general more fragile than single-layer systems [334]. Moreover, the total disruption of the network occurs with a discontinuous phase transition, which is characterized by large avalanches of failure events that propagate back and forth across the different layers of the network.

In this chapter, we provide an introduction to interdependent networks and review the main results about random percolation in multiplex networks. We then shift the focus to the problem of optimal multiplex percolation, presenting existing solutions, current challenges, and open research problems.

In the last section, entirely based on our work of Ref. [3], we present our contribution to the field. First, we disentangle the effect of overlap and inter-layer degree correlation on the optimal percolation problem, which was mostly neglected in previous works. We then provide novel evidence that the Pareto framework introduced in the previous chapters allows to generalise single-layer optimal percolation strategies to the case of multiplex networks. More importantly, we demonstrate that the Pareto-efficient approach can be effectively used to construct targeted strategies on correlated multiplex systems with overlap, which outperform all the existing methods in the case of synthetic duplex networks. Finally, we propose two new generalisations of a single-layer targeted strategy, which appear to outperform all the other targeted multiplex strategies when applied to real-world duplex systems.

6.1 Random percolation

The seminal paper of Buldyrev et al [334] introduced the notion of interdependent networks. In the multilayer network formalism, we consider two nodes as *interdependent* if the failure of one node in a layer also causes the failure of the same node on the other layers. The presence of interdependencies between the same node on different layers is usually represented by interlinks. In this chapter, we shall always refer to multiplex networks that are fully interdependent, i.e. all the nodes of the systems are interdependent.

In the classical percolation process on single-layer networks the robustness of the system is measured by the size of the giant connected component as a function of the damage inflicted. In analogy with the giant connected component in single-layer networks [13, 38, 335], we can evaluate the robustness of multiplex networks by computing the size of the so-called Largest Mutually Connected Component (LMCC) [or interchangeably the Mutually Connected Giant Component] as a function of random or targeted failures. Two nodes of a multiplex system $\mathcal{M}$ belong to the same Mutually Connected Component (MCC) if there exists at least one path

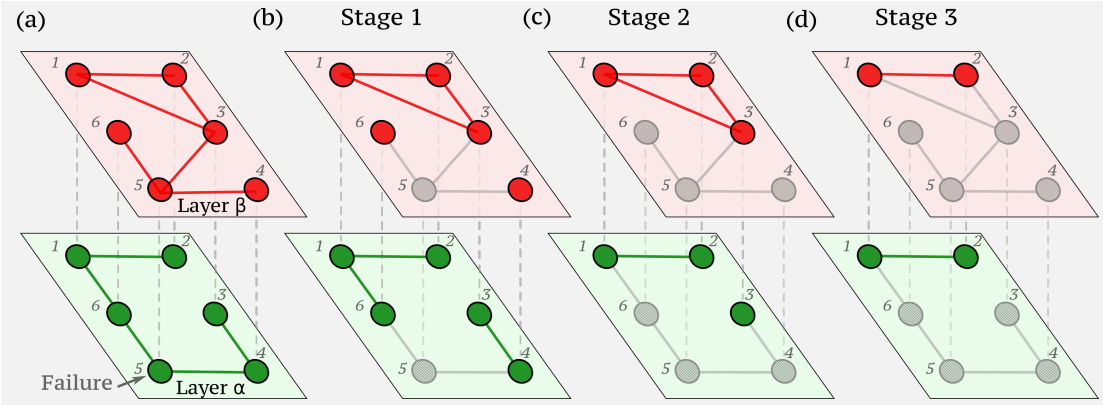

Figure 6.1. ***Schematic illustration of a cascade of failures in a duplex network.*** *The damage caused by the failure of a node in a duplex interdependent network propagates in both the layers. To compute the size of the LMCC after the removal of the node, we proceed using the steps indicated in the LMCC algorithm. That is, we remove all the links attached to that node in both layers (b). We then compute the connected components on both layers and iteratively remove links (marked as dashed gray) from a certain layer if the endpoints of such links belong to two different connect components in the other layer (c). We repeat the process of recomputing the components on each layer and removing links until no further damage is found. The remaining connected components are mutually connected. As consequence, the largest of these components represents the LMCC (d). Nodes that do not belong to the LMCC are indicated in grey.*

on each layer that connect them and traverses only nodes belonging to the same MCC. The LMCC represents the largest maximal sub-graph consisting of mutually connected nodes [336]. In particular, Buldyrev et al [334] have first shown how to compute the LMCC in multiplex networks as illustrated in Figure 6.1 for a multiplex with two layers. The algorithm proposed for determining the LMCC after an initial damage works along the following lines:

1. Compute the connected components on each layer $\alpha = 1, \ldots, M$ of the multiplex network
2. For each layer α, mark all links (i, j) whose endpoints belong to two different connected components in one of the other layers $\beta \neq \alpha$ as deactivated/removed.
3. If there are no new links marked as deactivated, then stop; otherwise restart again from point 1.

At the end of the algorithm, the remaining connected components are mutually connected. As a result, the largest of them represents the LMCC. Note that during the process, when all the links

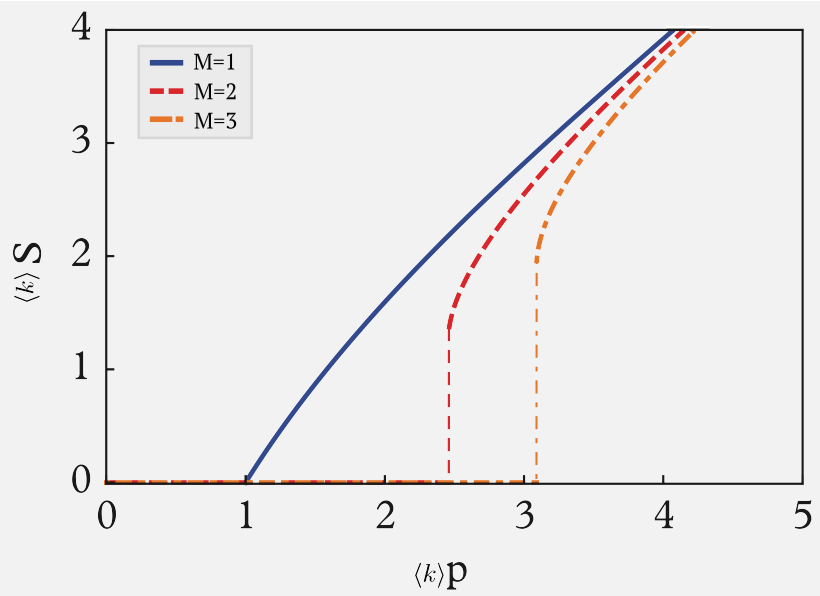

Figure 6.2. ***Percolation transition in single and multiplex systems.*** *We report the relative fraction S of nodes in the LMCC for multiplex networks with $M = 1, 2, 3$ layers and having Poisson degree distribution with identical average degree $\langle k \rangle$ on each layer as a function of the probability p that a node is not randomly damaged. Notice that for $M = 1$ the definition of LMCC coincides with the one of giant connected component, so that the phase transition is known to be continuous. To allow an easier comparison between the percolation transitions of the different multiplex networks, notice that the axes represent the product of the average degree $\langle k \rangle$ and the proportion (S/p). Adapted from [38].*

incident on a node i are marked as deactivated in all the layers (i.e. the node does not belong to the MCC), then the node i is considered as damaged/removed.

By analysing the relative size S of the LMCC as a function of the fraction of nodes removed from the multiplex, $f = 1 - p$, it is possible to determine the robustness of the system and the nature of the percolation transition. Here p is the fraction of nodes remaining in the multiplex. On the one hand, it has been shown that multiplex networks are usually much more fragile than the corresponding single-layer networks taken in isolation [38, 334]. This is because the failure of nodes in multiplex network usually propagates back and forth in the different layers and, as a result, it might trigger a cascade of failures in the system. On the other hand, the nature of the percolation transition in multiplex networks is dramatically different from the one appearing in single-layer networks. The authors of Ref. [334], in fact, have analytically shown

that the percolation transition in interdependent networks is a first order transition. This result is in stark contrast to the well-establish theory on single-layer networks, where the percolation transition occurs continuously [13, 19, 65] (see Fig. 6.2).

In light of these new results, an increasing amount of research has focused on: (i) characterising the nature of the percolation transition in multiplex networks as a function of the topology of each layer and the corresponding average degree; (ii) examining the impact of degree correlation and edge overlap on the LMCC; (iii) developing analytical formulations to determine the LMCC, as for instance, through message-passing algorithms [38, 229, 336, 337]; and (iv) relaxing the definition of LMCC to examine the case of multiplex networks with several layers [338]. Here, we provide a brief overview for some of these recent results. For example, it has been shown in Ref. [336] that multiplex networks are characterised by a hybrid percolation transition. In particular, the LMCC appears with a discontinuous transition for a given critical value p_c

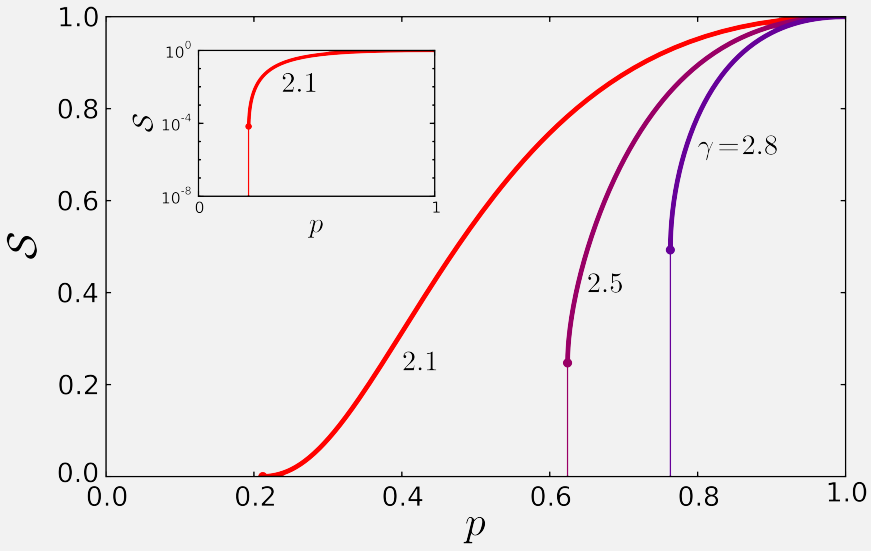

Figure 6.3. ***Effect of the power-law exponent on the robustness of duplex networks.*** *We report the relative size S of the LMCC as a function of the fraction of nodes p remaining undamaged for duplex networks whose layers have a scale-free distributions with exponent $\gamma = 2.8, 2.5, 2.1$ (from right to left) and same minimal degree. As the power-law exponent γ approaches 2, the height of the jump becomes smaller but is still non-vanishing, as it is possible to observe in the inset, which represents S vs p on a logarithmic vertical scale for $\gamma = 2.1$. Reprinted from [336].*

but it is also characterised by a square-root singularity from above. In other words, when the probability p that a node is not damaged approaches $p \to p_c^+$, a singularity appears. A more intriguing result, however, is provided in the same work and regards scale-free networks [13]. In particular, the authors of Ref. [336] observed that multiplex networks composed by layers with scale-free degree distributions with same average degree but different power-law exponent γ have a very peculiar behaviour. That is, as long as $\gamma > 2$, the percolation threshold p_c occurs with a (non-vanishing) first order transition that increases when γ decreases. In other words, multiplex systems whose layers are characterised by a broader scale-free distributions are in general much more prone to random damage than systems with a steeper scale-free degree distribution (see Fig. 6.3). An explanation for this phenomenon is that, on average, high-degree nodes (hubs) on a certain layer are coupled with low-degree nodes on the other layers and, as a consequence, the robustness of the system is heavily reduced. It is worth remarking that this result is in contrast to the behaviour observed for single-layer networks, where networks with scale-free distributions are usually very robust against random failures.

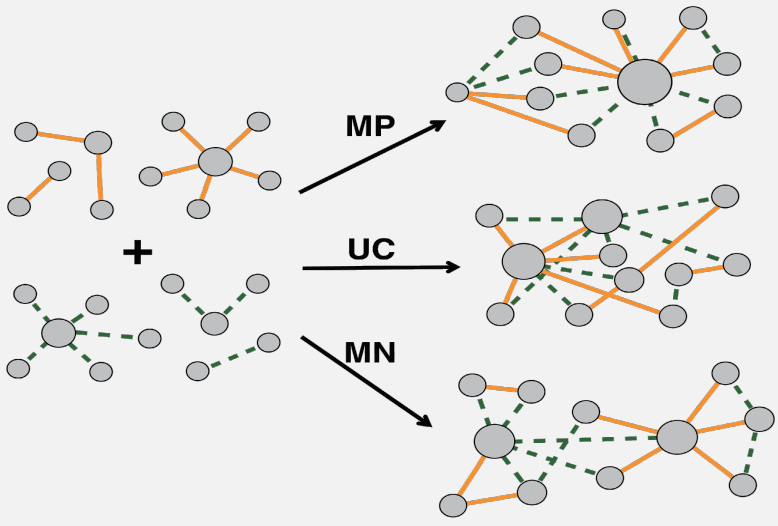

Figure 6.4. ***Schematic representation of three possible kinds of inter-layer degree correlations for duplex networks.*** *These correlations correspond to maximally positive (MP), uncorrelated (UC), and maximally negative (MN). Each layer of the duplex systems is represented by a different type of link, respectively indicated by the solid and dashed edges. Reprinted from [339].*

6.1.1 The role of inter-layer degree correlation

Several studies have also characterised the impact of inter-layer degree correlation in modulating the robustness of multiplex networks against random failures, observing that inter-layer degree correlation is responsible for substantial shifts in the percolation threshold p_c [339, 340]. In particular, one of the first studies was performed by Min et al. [339]. They considered the robustness of duplex systems for three representative correlated structures, namely, maximally positive (MP), uncorrelated (UC), and maximally negative (MN). An illustrative representation of these tree different structures is reported in Fig. 6.4.

These settings correspond to the following scenarios. When considering MN correlated systems, hub nodes are associated with low-degree nodes. By contrast, in MP multiplex networks, hubs are interdependent on hubs in the other layers. We report in Fig. 6.5 the relative size S of the LMCC as a function of the fraction of nodes p remaining undamaged for three different correlated duplex networks with Poisson degree distribution. Remarkably, when considering a sequence of random failures in the system, MN systems are much more fragile than MP multiplex networks. This is because high-degree nodes are essential in maintaining each layer connected but, in MN multiplex networks, these nodes can be damaged either as a result of a direct failure or by the failure of one of the correspondent interdependent low-degree nodes on the other layers.

6.1.2 The role of link overlap

As discussed in Chapter 2.2, several real-world multiplex networks are characterised by the presence of a non-negligible amount of link overlap. As a result, numerous studies started examining the percolation transition on these kind of systems, providing two approaches to investigate the percolation phase transitions [228]. The first one was proposed in Ref. [341, 342] and it is based on a coarse-grained decomposition of multiplex networks in terms of the so-called supernodes.

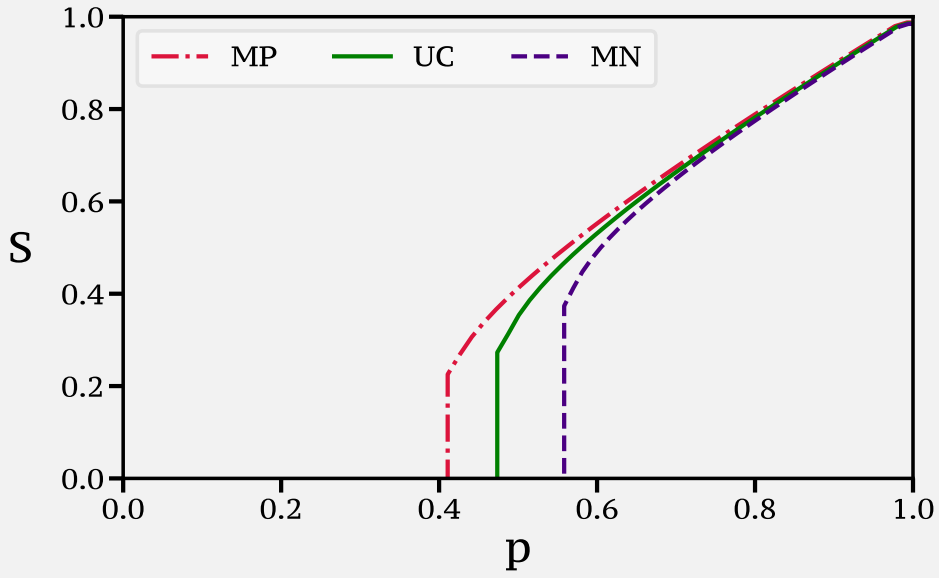

Figure 6.5. ***Impact of inter-layer degree correlation when considering random percolation on duplex networks.*** *We report the relative size S of the LMCC as a function of the fraction of nodes p remaining undamaged for duplex networks with three different levels of degree correlations, namely, Maximally Positive (MP), no degree correlation (UC), and maximally negative (MN). The duplex networks consist of two Poisson layers with $N = 10^4$ nodes and same average degree $\langle k \rangle = 5$.*

With this representation, it is possible to analytically characterise the percolation transition, but it only works for duplex systems. A more refined approach was proposed in Ref. [38, 229] using a message-passing algorithm, which allows to analytically characterise the percolation phase transition on multiplex networks with an arbitrary number of layers, as long as the local tree-like assumption remains valid. Remarkably, results presented in Ref. [343] have recently demonstrated how to create message-passing methods that work for any network, regardless of structure.

Independently from the analytical formulation considered, the main finding of these works is that link overlap is responsible for a substantial shift in the position of the percolation transition. However, the transition always remain discontinuous in all the cases, with the only exception represented by the case in which there is complete overlap of all the layers of the multiplex network (i.e. all the layers are identical). In that case, the LMCC of the multiplex network coincides to the largest connected component of a single layer network. Hence, the transition is

second order. Finally, in Ref. [340], the authors examined the combination of degree correlation and link overlap, providing new characterisation for the more complex phase transitions.

6.1.3 Analytical approximation through message-passing algorithms

Message passing techniques are widely used for characterising critical phenomena and dynamical systems in complex networks [344–346]. Recently, they have been successfully used in Ref. [344] to estimate the size of the percolating cluster and the average size of nonpercolating clusters in single-layer networks, providing also an analytical expression for the position of the percolation threshold in terms of the inverse of the leading eigenvalue of the non-backtracking matrix [347]. Other applications include network control [345] and epidemic spreading [346]. In the context of interdependent multiplex networks, the message-passing method was first proposed in Ref. [337], and subsequently applied on multiplex networks considering a variety of different conditions [38, 229, 340]. For instance, it has been used for (i) characterising the robustness of real multiplex networks and (ii) to study the percolation phase diagram of multiplex networks with link overlap and arbitrary number of layers.

According to this method, nodes send messages along their links to neighbouring nodes. These messages can be used to determine whether a node is in the MCGC or not. Note that this approach is valid on locally treelike random networks, for which mean-field theory based on generating functions becomes exact in the large system limit [337]. We refer to Ref. [38] for a detailed discussion on the topic.

6.2 Optimal percolation

As discussed in the previous sections, a large part of recent research has focused on studying the impact of randomly removing nodes from the multiplex network [228, 348–351] and on charac-

terising the wide spectrum of possible percolation transitions [228–230, 340, 342, 349]. However, quite often targeted attacks can potentially drive a system to collapse by knocking down a much smaller fraction of nodes than required by random attacks [352–355]. In this section, we shift the focus to optimal percolation, which is the problem of finding the minimal fraction of nodes whose removal would irreversibly fragment the system. The solution of the optimal percolation problem has a double-pronged importance. On the one hand, it has important implications for influence maximisation in opinion dynamics [356, 357] and for effective immunisation in spreading processes [358–360]. On the other hand, it represents the cheapest way to dismantle a network [361]. In the last five years, an increasing amount of research has focused on finding scalable heuristics to approximate the solution of the optimal percolation problem in single-layer network [362–365], which is known to be NP-hard [356]. Only recently, however, the same problem has been investigated in the context of multilayer networks [366]. Although there are currently no studies about the computational complexity of optimal multiplex percolation, it is reasonable to assume that this problem is not easier than it's classical single-layer counterpart.

In analogy with random percolation discussed above, also in the context of optimal percolation the parameter of interest corresponds to the relative size of the largest mutually connected component (LMCC). However, in this case, the final objective translates to finding the smallest set of nodes which, if removed, would reduce the size of the LMCC to $\mathcal{O}(N^{1/2})$. We refer to these nodes as the *critical nodes*, which constitute the *critical set* (alternatively the *attack set*). We denote the relative size of this set as q. In the work of Osat et al [366], the authors focused on examining the role of the multiplex structure when investigating the optimal percolation process. In particular, they compare the size of the critical set obtained in duplex networks with the critical set found for the single-layer representations extracted from the multiplex system (i.e. the individual layers or the aggregate representation). From random multiplex percolation theory [38], we know in advance that $q \leq q_{\text{monoplex}} \leq q_{\mathcal{A}}$, where q_{monoplex} and $q_{\mathcal{A}}$ respectively denote the relative size of the critical set for the single layers and for the aggregate network.

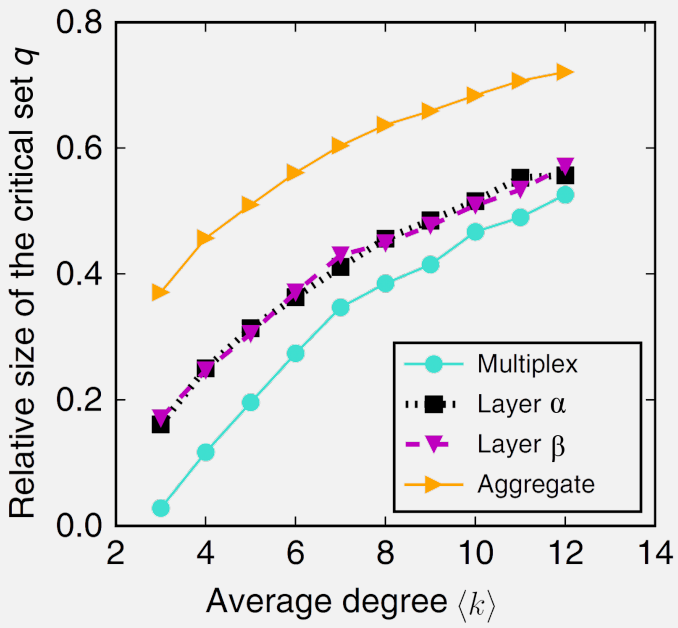

Figure 6.6. ***Optimal percolation on synthetic duplex networks.*** *The layers of the systems are two Erdös-Rényi random graphs with $N = 1000$ nodes and tunable average degree $\langle k \rangle$. The relative size of the critical set q of duplex network is reported as a function of the average degree $\langle k \rangle$ (turquoise circles), and compare it with the same quantity when computed on the individual layers (black squares and purple triangles) or the aggregate network (orange triangles). The size of the critical set is estimated using the SA algorithm. Adapted from the work of S. Osat, F. Ali and F. Radicchi [366].*

However, it is crucial to quantitatively characterise how poorly the quantity q_{monoplex} and $q_{\mathcal{A}}$ underestimate the actual robustness of the multiplex representation. We report in Fig. 6.6 such a comparison when considering the size of the structural set identified using the Simulated Annealing (SA) optimisation algorithm as a function of the average degree $\langle k \rangle$ and for synthetic duplex networks whose layers are Erdös-Rényi random graphs. Remarkably, for relative small values of $\langle k \rangle$, dismantling an ER duplex network requires knocking down a far smaller number of nodes than any of its constituent layers. Notice that E-R random graphs are usually considered in the optimal percolation literature (both single and multi-layer networks [362, 365–367]) as a baseline for investigation.

In the same work by Osat et al [366], the authors also investigate the role of overlap and inter-layer degree correlations, which are known to have important effects on random percolation [339,

340], as presented before. They started from two identical Erdös-Rényi random graphs with $N = 1000$ nodes and $\langle k \rangle = 5$ and considered a very simple model to simultaneously change edge overlap and the inter-layer degree correlations [368]. That is, nodes in one of the two layers are subject to relabelling with a certain probability ζ. When $\zeta = 0$, duplex, aggregate network and both the single-layer networks are identical. By contrast, when $\zeta = 1$, the layers of the duplex network correspond to two Erdös-Rényi random graphs with identical degree sequence and negligible overlap. As ζ increases, both edge overlap and inter-layer degree correlation simultaneously decrease in a non-trivial way. We report in Fig. 6.7 the results of this analysis. Clearly, this simple model cannot disentangle the role played by edge overlap and by the inter-layer degree correlations, and yet it is evident that the combination of these two factors sensibly affects the robustness of the duplex systems.

Finally, the authors of Ref. [366] observe that most of the single-layer optimal attack strategies [360, 362–364] cannot be easily extended to the multilayer case. This is because most of the heuristic algorithms proceed by assigning a score to each node, based on some structural indicator [11, 65, 365], and then iteratively removing nodes in decreasing order of their score. However, in a multiplex network, it is not immediate how to combine node scores on different layers to obtain a meaningful ranking. For example, single-layer methods that use node degrees as ingredients to calculate and assign scores to each of the nodes cannot be directly mapped to duplex networks. This is because a node in a multiplex network has multiple degree values. Hence, it is not clear what is the best way of combining these values to obtain a single score. Or alternatively, let us consider the CoreHD algorithm [364], which is one of the most effective strategies to destroy the giant connected component of a single-layer graph. The algorithm proceeds by iteratively removing the nodes with the highest degrees from the 2-core of the graph (i.e., by effectively de-cycling the network). However, this idea cannot be directly applied to duplex networks, since the 2-core of a multiplex graph is not uniquely defined. As a consequence, there are several existing multiplex extensions of the CoreHD strategy, but none of them provides

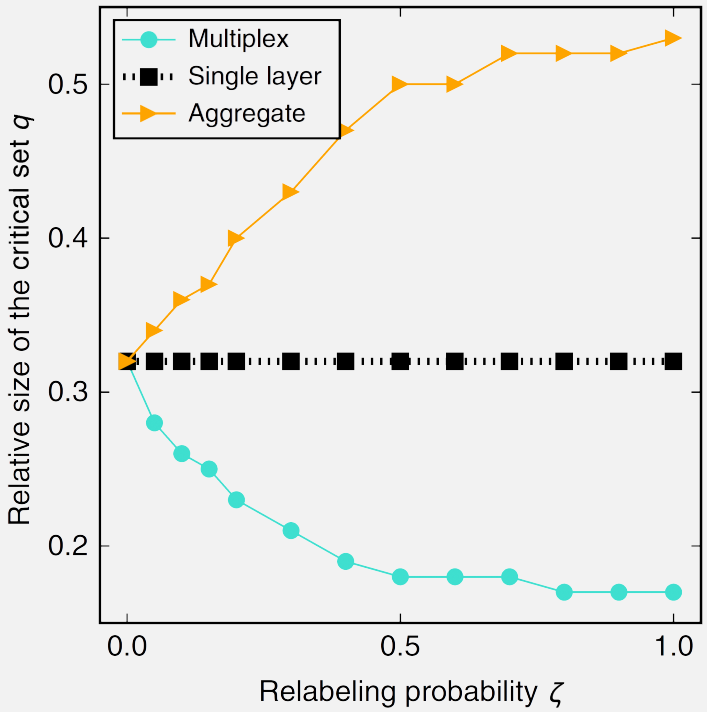

Figure 6.7. ***Joint effect of edge overlap and inter-layer degree correlation on the size of critical sets***. *The authors of Ref. [366] considered a duplex network consisting of two identical Erdös-Rényi layers with* $N = 1000$ *and* $\langle k \rangle = 5$. *Using the model presented in Ref. [368], in one of the two layers, the labels of the nodes are randomly reshuffled with probability* ζ. *The mean value of the size of the critical sets obtained using the SA algorithm and averaged over* 100 *realisations is respectively reported for the single layers, aggregate and duplex networks. Adapted from [366].*

satisfactory results on duplex networks [367]. It is worth remarking that despite the wide usage of the Simulated Annealing algorithm in the results shown above, the SA has two main drawbacks. That is, *(i)* it is prohibitively slow for large system sizes and therefore it cannot be employed for practical purposes, *(ii)* it does not provide any insight related to the structure of a good critical set. Hence, relying on heuristics that closely approximate the solution of the optimal multiplex percolation problem becomes the only feasible approach for estimating the critical set of large-scale systems.

The recently proposed heuristic Effective Multiplex Degree (EMD) [367] represents the state-of-the-art targeted attack strategy. The underlying idea of the EMD heuristic is based on taking into account the multilayer structure of the system, including inter-layer adjacency. In this way,

the EMD scores effectively exploit the degree-heterogeneity between different layers. In the case of a duplex network, the EMD score of node i is defined as [367]:

$$w_i = \sum_{\alpha} \sum_{j \in \mathcal{N}_i^{[\alpha]}} \frac{w_j}{k_j^{[\alpha]}} \tag{6.1}$$

where $k_j^{[\alpha]}$ is the degree of node j at layer α, $\mathcal{N}_i^{[\alpha]}$ represents the set of neighbours of node i at layer α, and w_i is the EMD weight, which accounts for the impact on the LMCC when removing the node i. Notice that Eq. (6.1) is a self-consistent equation, so that higher EMD weights are associated with nodes with larger amount of connections, that at the same time are also connected to neighbouring nodes having high EMD weights.

The weight w_i represents the equilibrium probability to find a random walker at node i, when the random walker is allowed to switch uniformly between layers and select edges within that layer uniformly at random. As a consequence, Eq. (6.1) may be seen similar to a PageRank [369] summed over layers, however, it does not involve any random jump/teleportation/dumping parameter α that allow walkers to randomly jumps across the system. In particular, to obtain the values of the EMD weights, the authors of Ref. [367] set their initial values equal to the sum of the degrees in all layers, i.e. $w_i^{(0)} = \sum_\alpha k_i^{[\alpha]}$. They then update the weights using Eq. (6.1) until the weights fully converge, i.e. the largest relative difference between $w_i^{(t)}$ and $w_i^{(t-1)}$ is less than $\varepsilon = 10^{-7}$. The EMD method uses an adaptive strategy, so that the ranks are recalculated after each node removal from the duplex network.

We report the performances of the EMD heuristic in Fig. 6.8 for a duplex network whose layers are two independent realisation of Erdös-Rényi random graphs with $N = 10^5$ nodes and $\langle k \rangle = 5$. For comparison, along with the results of the EMD heuristic, we also report other "simple" generalisations of single-layer targeted attack strategies. These methods correspond to: *(i)* the sum of the rankings in all the layers of the collective influence propagation (CIp_S) algorithm [370]; *(ii)* the generalisation of the CoreHD method obtained by considering the highest

sum of degrees within the union of the 2-cores of all layers (CoreHDs) or by considering nodes with the highest degree within the 2-core of the layer that has the lowest mean degree (CoreHD1); *(iii)* methods obtained by removing the node with the highest degree sum (DegSum) or product (HDA). It is worth noticing that all these methods are adaptive, which means that after the removal of a node from the system, all of the scores of the remaining nodes are recomputed. As previously announced and shown in Fig. 6.8, simple generalisations of targeted strategies originally designed for single-layer networks have very poor performance when applied to duplex systems. However, this is not really surprising, since these generalisations cannot capture the presence of interdependencies in duplex systems, which is instead crucial when investigating the robustness of multiplex systems.

Another important aspect that we need to focus on is that the results reported in Fig. 6.8

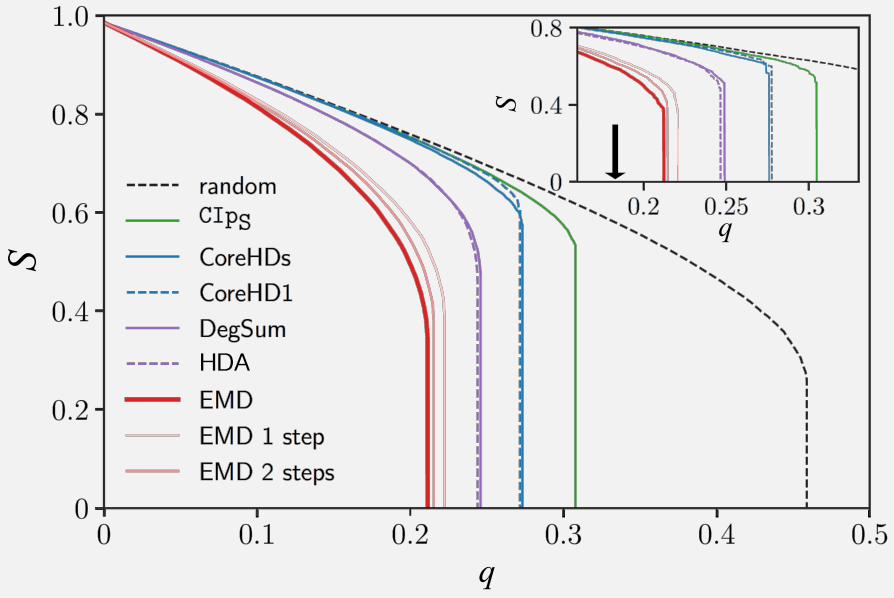

Figure 6.8. ***Performance of the EMD strategy for duplex networks.*** *We report the relative size S of the LMCC of a multiplex network when a fraction q of nodes is removed by different targeted attack strategies. The multiplex consists of two independent realisation of Erdös-Rényi random graphs with $N = 10^5$ nodes and $\langle k \rangle = 5$. From left to right the methods reported are: EMD and the approximate solutions of Eq. (6.1) using only one or two iterations, highest degree product (HDA) and sum (DegSum), CoreHD1, CoreHDs, the collective influence propagation (CIp_S), and random damage (random). Results for duplex systems with $N = 10000$ nodes and same average degree are reported in the inset plot. The critical set identified by the simulated annealing algorithm is reported for comparison (black arrow). Adapted from [367].*

hold for duplex system with negligible overlap and uncorrelated inter-layer degree correlations (i.e. the two layers are independent realisations of the ER model). However, correlations and overlap are a salient aspect of all real-world multiplex networks [38, 224, 339, 371] and, as shown in Fig. 6.7, the interplay of those two quantities substantially affect the size of the critical set q. As a consequence, it is fundamental to disentangle the role played by these two quantities when we aim at accurately estimating the size of the critical set. In principle, we have no guarantees that the EMD heuristic, or any other method, performs well when applied to duplex systems with different conditions. Indeed, as we shall see in the next sections in a more quantitative way, disentangling the effect of these two quantities is fundamental to accurately determine the robustness of duplex systems in the context of optimal percolation.

As discussed in the previous sections, one of the first studies investigating the impact of inter-layer degree correlation when estimating the robustness properties of duplex networks was still provided by Min et al. [339]. However, along with the results reported for the random percolation theory, they considered the robustness of duplex systems against targeted attack (i.e. removing nodes with the highest degree in the duplex) when the systems has negligible overlap and for the same three representative correlated structure reported in Fig. 6.4.

Interestingly, for duplex systems consisting of Erdös-Rényi random graphs with moderate average degree (approximatively more than 4), they found that maximally negative correlated systems are more robust against targeted attack on high-degree nodes than duplex systems with a maximally positive correlated structure. However, as we shall see in the next sections, if we blindly generalise this result to the optimal percolation context, we could end up with misleading conclusions. In fact, duplex networks with maximally negative inter-layer degree correlation are hyper-fragile compared to systems with maximally positive degree correlation only when the system is characterised by negligible overlap.

6.3 Optimal percolation in correlated multiplex networks with overlap

Although correlations and overlap are indeed a salient aspect of all real-world multiplex networks, the few strategies for optimal multiplex percolation proposed so far have been mainly tested on synthetic uncorrelated multilayer networks, thus neglecting interlayer degree correlations and edge overlap. In this section, we focus on studying the optimal percolation problem for duplex systems with non-trivial inter-layer degree correlations and non-negligible edge overlap. First, we introduce two new classes of algorithms, respectively based on a generalisation to duplex networks of the Collective Influence algorithm [362], and on the concept of Pareto efficiency presented in Chapter 4, which allows to combine layer-based and genuinely multi-layer node properties. Second, we disentangle the effect of inter-layer degree correlations and edge overlap when estimating the robustness of systems under targeted attacks. In particular, all the existing algorithms for optimal percolation systematically overestimate the size q of the critical set of nodes to knock down in order to destroy the mutually connected giant component. Finally, we show through extensive numerical simulations that the Pareto efficient targeted attack strategies identify consistently smaller critical sets in synthetic correlated multilayer networks. By contrast, the generalisations of Collective Influence outperform several state-of-the-art algorithms in real-world duplex systems.

6.3.1 Duplex Collective Influence

An efficient heuristic for optimal single-layer percolation was introduced in Ref. [362] by Morone and Makse. The authors mapped optimal percolation into the minimisation of energy of a many-body system, in which the interactions among units are expressed in terms of the non-backtracking matrix of the graph [347], and proposed an efficient and scalable algorithm, called

Collective Influence (CI), to identify the minimal set of influential nodes to remove. The CI algorithm iteratively removes nodes according to the highest values of CI scores, defined as [362]:

$$CI_\ell(i) = (k_i - 1) \sum_{j \in \partial\mathcal{B}(i,\ell)} (k_j - 1) \tag{6.2}$$

where k_i is the degree of node i, $\mathcal{B}(i,\ell)$ is the set of nodes inside a ball of radius ℓ (defined as the shortest path) around node i, while $\partial\mathcal{B}(i,\ell)$ represents the frontier of the ball of radius ℓ containing all the nodes at distance smaller than or equal to ℓ from node i. This means that a node i is assigned a larger CI score if the set of nodes at distance ℓ from i has a large number of links. By removing a node with a large CI score, we are potentially removing a node that mediates a large number of walks. Notice that the factor $k_i - 1$ comes from the minimisation of the cost energy function described in terms of the non-backtracking matrix of the graph (see Supplementary Note IIE of Ref. [362] for the analytical derivation). Remarkably, the attack strategy based on CI can be implemented by an algorithm with time complexity $\mathcal{O}(N \log N)$, which is attained by using a max-heap to keep and update the CI scores of nodes [370]. Some variations of the CI heuristic have managed to obtain relatively better performance (i.e., smaller attack sets) by including more structural information about the relevance of a given node for percolation [370], at the cost of an increased time complexity. As we have shown in the previous section, there has also been an attempt to extend the Collective Influence algorithm to the case of duplex networks by combining the bare CI scores of the nodes at the two layers [367], but the results are not competitive with other existing algorithms. The main reason is that the bare combination of the layer-based scores does not take into account the role played by edge overlap and inter-layer degree correlations in triggering a cascade of node removals.

We introduce here two generalisations of Collective Influence for duplex networks, which automatically take into account both inter-layer degree correlations and edge overlap. The heuristics are based on two simple ideas: the first one is that nodes with high degrees and high edge overlap

are more likely responsible for mediating a lot of interdependent paths; the second one is that the removal of a given node i has a large impact on the size of the MCC if it triggers a larger cascade of node removals *away* from i. We define the Duplex Collective Influence (DCI) as follows:

$$DCI(i) = \frac{k_i^{[1]} k_i^{[2]} - k_i^{\text{int}}}{k_i^{\text{aggr}}} \left[\sum_j a_{ij}^{[1]} (k_j^{[2]} - 1) + a_{ij}^{[2]} (k_j^{[1]} - 1) \right] \tag{6.3}$$

where k_i^{int} is the degree of node i in the intersection graph (i.e., the graph containing only the links which appear in both layers), and k_i^{aggr} is the degree of node i in the binary aggregated graph. The DCI score of a given node i is indeed obtained as the product of two terms. The first contribution is due to the product of the degrees of node i at the two layers and to the local edge overlap of node i. It is easy to show that this term increases when k_i^{int} increases, meaning that nodes with a high edge overlap and high degrees a the two layers will in general be ranked higher. The term in square brackets, instead, takes into account potential cascades away from node i triggered by the removal of i. In particular, the term is larger if the neighbours of i on layer 1 have a high degree on layer 2, and vice-versa. In this case, the removal of i (and of all its edges on both layers) will disrupt all the paths between the neighbours of i on layer 2 which are mediated by i, hence potentially disrupting the connected component to which i belongs at layer 2. This might in turn trigger further node removals in the neighbourhoods of those nodes, and let the cascade propagate away from node i. In the limiting case of a duplex network consisting of two identical layers (which is indeed equivalent to a single-layer network with respect to percolation), DCI yields the same node ranking as that induced by CI on the aggregated network when we set $\ell = 1$ in Eq. (6.2). We report in Section 6.4 the additional details.

It is important to note that when nodes are iteratively removed from a duplex, the term $k_i^{[1]} k_i^{[2]}$ in Eq. (6.3) might become equal to zero, e.g., due to the removal of nodes around i which have left node i isolated in one of the two layers. However, node i might still have a relatively large degree on the other layer, and its removal might trigger larger cascades away from i than a

node which is still connected on both layers but has a small degree on each of them. This happens more frequently in duplex networks with heterogeneous degree distributions. To account for this inconvenience, we define a modified DCI score:

$$DCI_z(i) = \frac{(k_i^{[1]}+1)(k_i^{[2]}+1) - 3k_i^{\text{int}} - 1}{k_i^{\text{aggr}}} \left[\sum_j a_{ij}^{[1]}(k_j^{[2]}-1) + a_{ij}^{[2]}(k_j^{[1]}-1) \right] \tag{6.4}$$

which is obtained by replacing $k_i^{[\alpha]}$ with $k_i^{[\alpha]}+1$ in Eq. (6.3), and enforcing that DCI_z induces the same node ranking as CI with $\ell = 1$ in the limiting case of duplex networks made of two identical layers. The subscript z indicates that we are correcting for nodes with zero degree on at least one of the two layers.

We use DCI and DCI_z in an adaptive algorithm that iteratively removes nodes from the duplex according to their score recomputed on the remaining sub-graph. This process is iterated until the size of the LMCC becomes non-extensive i.e. $\mathcal{O}(N^{1/2})$. The time complexity of the direct implementation of this algorithm by means of simple data structures is $O(N^2 \log N)$, but a more efficient algorithm which uses a max-heap to keep the list of scores sorted will have time complexity $O(N^{1.2})$. We provide all the details on the time complexity in Section 6.4.3.

6.3.2 Pareto efficient targeted attacks

We propose a second class of attack strategies based on the hypothesis that it should be possible to obtain smaller attack sets by combining layer-specific and genuinely multilayer information. We use here the concept of Pareto efficiency presented in Chapter 4. The idea is illustrated in Fig. 6.9. We consider a set of m node descriptors, which we deem relevant for multilayer percolation, so that each node i is associated with the vector of ranks induced by each of the m scores $\boldsymbol{r}^i = [r_1^i, r_2^i, \ldots, r_m^i]$, and is mapped into a point of an m-dimensional space C. Assuming that optimal attack sets consist of nodes who are maximising all the structural descriptors at the same time (i.e. the higher the rank r_t^i of node i, the more important is node i for the structural descriptor t), then we employ the concept of dominance strict partial order to identify

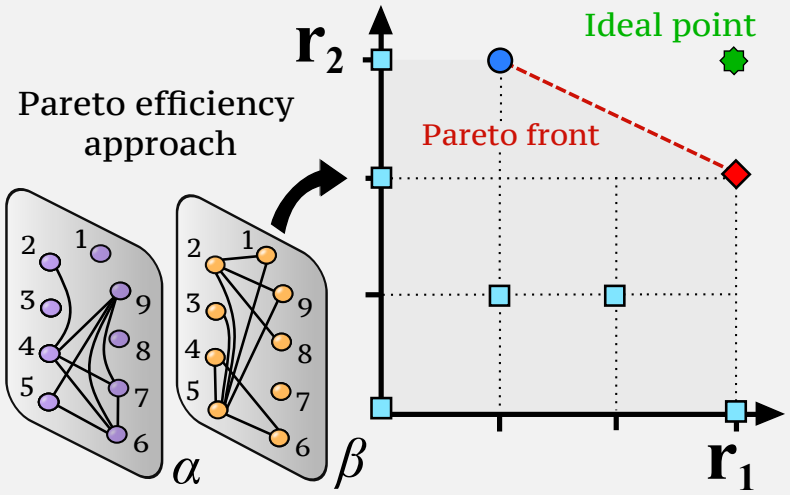

Figure 6.9. ***Graphical representation of Pareto efficiency for two generic structural node descriptors r_1 and r_2.*** *Each node of the multiplex is mapped onto a point in the (r_1, r_2) plane. The points for which no improvement can be achieved in one objective function without hindering the others are called Pareto-efficient (blue circle and red diamond) and constitute a Pareto front. The Pareto-efficient points are iteratively ranked according to their Euclidean distance from the ideal point (green star), i.e., the point that maximises all the objective functions. In this case, the node associated with the red diamond is ranked first.*

Pareto-efficient nodes in the space C.

At a first glance, the Pareto-efficiency approach might appear similar to the hybrid methods presented in Ref. [372], however, there are a few fundamental differences. Indeed, as presented in Chapter 4, the Pareto-efficiency approach: *i)* is agnostic with respect to the functions to be maximised (i.e., it is parameter–free); *ii)* it has a simple physical interpretation (i.e., multi-objective optimisation arises naturally whenever a system is subject to at least two concurrent sets of constraints); and *iii)* is known to have several advantages over scalarisation methods [267, 281].

However, in this context, the main drawback of the Pareto-efficient approach is that it proposes a set of equally-viable solutions at each iteration, and such a set normally contains multiple solutions. This is indeed far from ideal, since comparing the performance of different cost functions (obtained from different ways of ranking nodes on the basis of their structural properties) can become somehow complicated. However, there are many ways to select only one of the Pareto-optimal solutions from a Pareto front, when no additional information is available about

how preferable a certain solution is. Here, we consider the solution that has the minimal Euclidean distance from the ideal point (see Chapter 4 for a formal definition). In other words, for each Pareto strategy, we constructed the critical sets by iteratively removing the Pareto-efficient point having minimal Euclidean distance from the ideal point (potential ties are broken by selecting one of the points uniformly at random). We then recompute the set of Pareto-efficient points and iterate until the LMCC becomes non-extensive.

6.3.3 Comparison of targeted attack strategies

Here we compare the algorithms for optimal multiplex percolation proposed so far, namely the Highest Degree product Adaptive (HDA) [366] and Effective Multi-Degree (EMD) [367], with a variety of multiplex targeted attack strategies from three classes, namely *i)* alternative genuinely multiplex strategies; *ii)* Pareto-efficient strategies based on the combination of the scores of single-layer targeted attack strategies on the two layers; and *iii)* Pareto-efficient strategies obtained by combining single-layer descriptors with one genuinely multiplex algorithm.

In the following we will discuss in detail the performance obtained by six strategies, namely HDA, EMD, DCI , DCI_z , and the two Pareto-efficient strategies obtained by combining the k-core ranking on the two layers with the ranking induced by EMD, that we call (k-core, EMD), and the score assigned on each layer by Belief Propagation Decimation [363] and the ranking induced by DCI_z . As a reference, we also report the results obtained by Simulated Annealing (SA) as described in Ref. [367], which is able to find very small targeted attack sets at the expense of relatively heavier computations. All the results presented in this section containing the comparisons with all the other methods we tested are reported in Section 6.5.

In Fig. 6.10(a), we report the percolation diagrams of duplex networks with uncorrelated Erdös-Rényi layers. Notice that the duplex network in Fig. 6.10(a) is consistent with that used in Ref. [367] and reported in Fig. 6.8 where the authors showed that the critical set found by

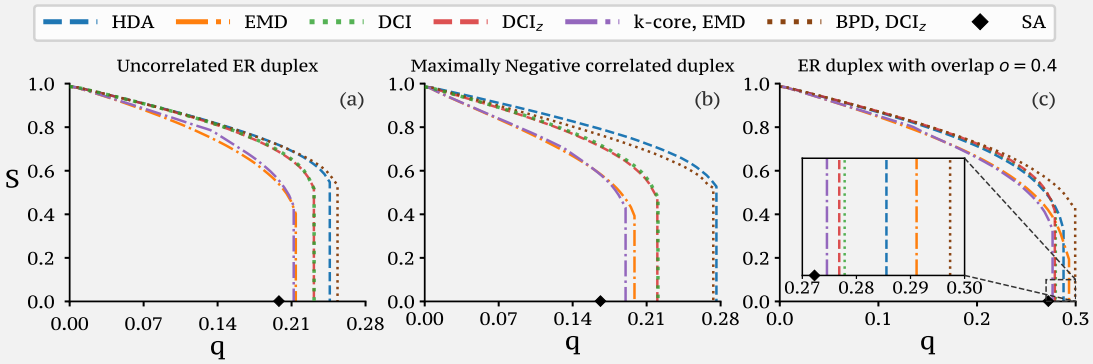

Figure 6.10. ***Performance of targeted attack strategies in duplex networks.*** *(a) Relative size S of the LMCC of a multiplex network when a fraction q of nodes is removed by different targeted attack strategies. The multiplex consists of two Erdös-Rényi layers with $N = 10^4$ nodes, average degree $\langle k \rangle = 5$, no inter-layer degree correlations ($\rho \approx 0$) and no edge overlap ($o \approx 0$). In this case, EMD provides a much smaller attack set than HDA, but the Pareto-efficient (k-core, EMD) strategy produces a smaller critical set. (b) If the two layers have maximally disassortative inter-layer degree correlations ($\rho = -1$) but still no edge overlap ($o \approx 0$), HDA performs sensibly worse than in the uncorrelated case, while the Pareto-efficient (k-core, EMD) finds a smaller attack set and again outperforms EMD. (c) If the duplex has a substantial edge overlap ($o = 0.4$) and no correlations ($\rho \approx 0$), the critical set is much larger than in the other two cases. The presence of edge overlap favours HDA, but the smallest attack set is still found by the (k-core, EMD) Pareto-efficient algorithm, immediately followed by DCI and DCI$_z$. For comparison, we also report in each plot the results obtained by Simulated Annealing (black diamond). All the curves are averaged over 20 realisations.*

EMD is usually much smaller than that found using HDA. Interestingly, the combination of EMD and k-core provides a smaller critical set than either EMD or HDA alone. This is because by targeting nodes which have high EMD scores and, at the same time, belong to the inner k-core on each layer, we have a higher probability of simultaneously damaging the LMCC of the multiplex and the giant connected component on each layer. Even more interesting results are reported in Fig. 6.10(b) for a duplex with maximally disassortative inter-layer degree correlations (and no edge overlap), and, respectively, in Fig. 6.10(c) for a duplex with high edge overlap. It is evident from the figures that the relative performance of each targeting algorithm depends quite substantially on the structure of the multiplex, and in particular on the presence of inter-layer degree correlation and edge overlap. For instance, EMD still outperforms HDA by a large margin when the multiplex has no edge overlap and disassortative degree-degree correlations [Fig. 6.10(b)], while EMD is the worst-performing strategy when edge overlap is not negligible.

In general, the algorithms based on Pareto-efficiency perform better than either EMD and HDA, while both DCI and DCI_z find relatively smaller critical sets in the case of networks with non-negligible overlap. This is a first confirmation that heuristics that perform better in one specific condition (e.g., where the two layers are uncorrelated and edge overlap is negligible) do not always achieve the same performance under other conditions.

6.3.3.1 Dependence on edge overlap

Here, we investigate the impact of edge overlap on the performance of different targeted attack strategies by considering a class of synthetic duplex networks with tunable edge overlap o. We consider the same definition of overlap used in Chapter 3.3.1 when analysing the complexity of multiplex networks. In the case of a duplex network, this quantity corresponds to the fraction of edges that are present on both layers, i.e. :

$$o_s = \frac{\sum_{i,j}^{N} o_{ij}}{2\sum_{i,j}^{N} \Theta\left(o_{ij}\right)} \tag{6.5}$$

where $o_{ij} = \sum_{\alpha=1}^{2} A_{ij}^{[\alpha]}$ and $\Theta(\bullet)$ is the Heaviside step function. We also consider the linear transformation $o = 2(o_s - 1/2)$ to map the edge overlap o_s into the interval $[0, 1]$.

To tune the overlap in duplex systems, we employ the model introduced in Chapter 5.2.1, which starts from two identical layers ($o = 1$ and maximal inter-layer degree correlation, $\rho = 1$) and iteratively rewires the edges of one of the two layers to reduce the edge overlap until we get to $o = 0$, yet maintaining untouched the degree sequence of each layer. In Fig. 6.11(a), we plot the relative size of the critical set q obtained by each of the six algorithms as a function of the edge overlap in a duplex with Erdös-Rényi layers. We notice that in general q is an increasing function of o. This fact is somehow expected from random percolation theory. In the limit case of a duplex with $o = 1$, the system is indeed indistinguishable from the single-layer graph obtained by combining the two (identical) layers, hence the optimal attack set in that case corresponds

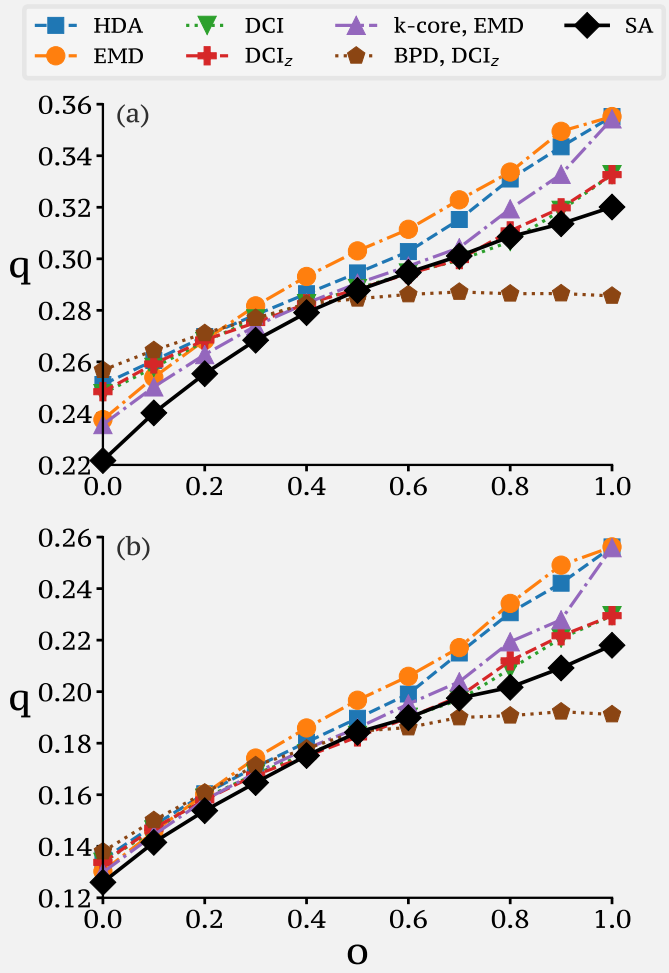

Figure 6.11. ***Impact of overlap for the optimal percolation problem***. *We report the size of the critical attack set q as a function of edge overlap for different attack strategies in duplex networks with $N = 10^4$ nodes, $\langle k \rangle = 5$, and whose layers have (a) Erdös-Rényi or (b) scale-free degree distributions ($\gamma = 2.6$). The plots are obtained by starting from two identical layers ($o = 1$ and $\rho = 1$) and then iteratively rewiring the edges of one of the two layers to reduce the edge overlap until we get to $o = 0$ [41]. Again, strategies based on Pareto-efficiency yield the best results. Results averaged over 20 realisations. Error bars are smaller than the marker size.*

to that of each layer. However, each attack strategy behaves slightly differently as o increases. For instance, for $o > 0.3$ the critical set found by EMD is always larger than that obtained by all the other strategies. In the limit of $o = 1$, however, the EMD and HDA heuristics coincide, since the EMD weight of each node i becomes proportional to the degree k_i. By contrast, DCI , DCI$_z$ and the two Pareto-efficient strategies perform relatively poorly in networks with small overlap, but they generally outperform both EMD and HDA as the amount of overlap increases.

This is because targeted methods that indirectly disrupt a large number of interdependent paths are more likely to trigger cascades in the system.

Notice that some Pareto-efficient strategies already outperform the results of the Simulated Annealing achievable in a reasonable computing time (same implementation as the one presented in Ref. [367] with temperature steps equal to 10^{-7}). A similar qualitative behaviour is observed when considering duplex systems having heterogeneous degree distribution on each layer [Fig. 6.11(b)], although the typical values of q are overall smaller. This indicates that the heterogeneity of the degree distribution of each layer has some impact on the efficiency of each attack strategy, but the presence of edge overlap effectively determines the relative performance of different strategies. Interestingly, for both the topologies, the best (smallest) critical set is always obtained by one of the methods we proposes, that is, heuristics that combine layer-based and genuinely multilayer node properties through Pareto-efficiency, with DCI and DCI_z following closely when $o > 0.4$.

6.3.3.2 The role of inter-layer degree correlations

Here, we use the procedure explained in Chapter 5.2.3 to tune inter-layer degree correlations between the maximally disassortative case [or alternatively Maximally Negative (MN)] and the maximally assortative one [Maximally Positive (MP)]. In order to isolate the effect of inter-layer degree correlations, we study the performance of the six targeted attack strategies as a function of the inter-layer degree correlation coefficient ρ [224], imposing that each realisation of the multiplex had $o \approx 0$. To simultaneously account for the joint effect of overlap and inter-layer degree correlations, we also consider the sequences of multiplex networks obtained by increasing ρ while keeping the edge overlap fixed at a given value. To obtain those sequences, we implemented a new procedure, in which we first increase the value of inter-layer degree correlation [224], and then we set the desired value of edge overlap through the procedure introduced in Chapter 5.2.1 based on biased edge rewiring [2, 41].

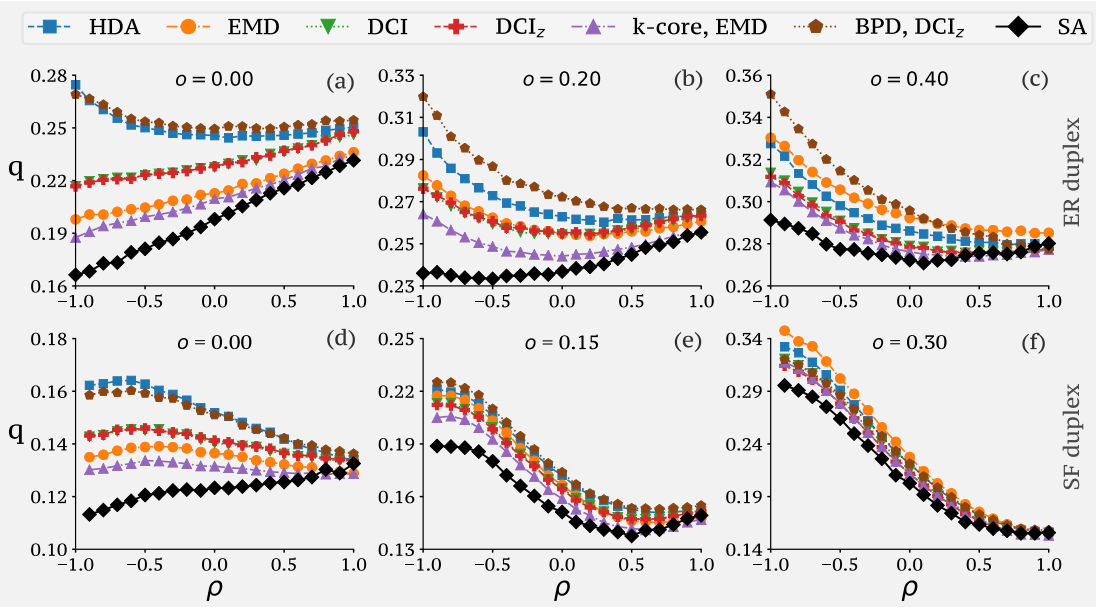

Figure 6.12. ***Impact of inter-layer degree correlation and overlap in the optimal percolation problem***. *We report the critical set q as a function of the inter-layer degree correlations coefficient ρ [224] for duplex networks with $N = 10^4$ nodes, whose layers are (a-c) Erdös-Rényi graphs with $\langle k \rangle = 5$, and (e-f) scale-free networks with $\gamma = 2.6$ (e-f). For each topology, we respectively report three different overlap conditions: (a, d) no edge overlap; (b, e) low edge overlap; (c, f) moderate edge overlap. The concurrent presence of inter-layer degree correlations and edge overlap strongly affects the robustness of a system against targeted attacks. This is particularly evident when considering duplex networks with heterogeneous degree distributions on each layer (d-f). When the edge overlap is non-negligible, duplex networks with maximally negative degree correlations are extremely robust under targeted attacks with respect to their maximally positive counterparts. Conversely, when the overlap is negligible, disassortatively-correlated duplex networks are more fragile. Overall, DCI , DCI$_z$, and the Pareto-efficient strategies that simultaneously combine single and multi-layer attacks, consistently detect smaller critical sets. The results obtained by Simulated Annealing (SA) are reported for comparison. Results averaged over 20 realisations. Error bars are smaller than the marker size.*

In Fig. 6.12 we plot the size of the critical set q identified by the six targeted attack strategies as a function of the inter-layer degree correlations ρ and for different values of edge overlap o. We report the results obtained on duplex networks with Erdös-Rényi layers [Fig. 6.12(a-c)], and with scale–free layers [Fig. 6.12(d-f)]. Interestingly, in all the scenarios considered EMD and HDA are outperformed by one or more of the heuristics we have introduced. In particular, the smallest critical set is often obtained by the (k-core, EMD) Pareto-efficient strategy. However, depending on the interplay between edge overlap and inter-layer degree correlations, profound differences among the six methods emerge. For instance, when considering a duplex with Erdös-

Rényi layers and negligible edge overlap [Fig. 6.12(a)], the discrepancy between the overall best strategy (k-core, EMD) and the worst one (HDA) is maximal when $\rho \simeq -1$. In particular, the critical set found by (k-core, EMD) when $\rho \simeq -1$ is around 19% (smaller than the one found for $\rho \simeq 1$, i.e., around 23%), while HDA finds a much larger critical set (28%), which is even larger than the one it finds for $\rho \simeq 1$ (25%). As a consequence, the presumed increased robustness of multiplex networks with disassortatively correlated degrees (mentioned in the work of Ref. [339]) is probably just an artefact of the algorithm used to determine the critical set [339]. By looking at the size of the critical set found by Simulated Annealing in Fig. 6.12(a), it seems clear that negatively-correlated multiplex systems without overlap are generally hyperfragile compared to positively-correlated ones. However, some of the attack strategies considered, including HDA and especially in uncorrelated systems, provide a diametrically opposite picture, and suggest that in absence of edge overlap positively correlated degree sequences are more fragile.

The results shown in Fig. 6.12(b)-(c) shed light on the interplay between edge overlap and inter-layer correlations. In both cases, the sizes of the critical sets found by the six algorithms are higher than those shown in Fig. 6.12(a) (i.e., when the edge overlap is negligible). In particular, the (sub-)optimal critical set q found by Simulated Annealing reveals that both edge overlap and inter-layer degree correlations contribute to determine the robustness of a duplex system.

Similar conclusions can be drawn by examining duplex networks with scale–free degree distribution [see Fig. 6.12(d-f)]. Also in this case both edge overlap and inter-layer degree correlations have a substantial impact on the performance of each algorithm. However, the difference between maximally-negative and maximally-positive correlated duplex networks becomes more relevant when edge overlap increases, mainly due to the fact that degree heterogeneity on each layer has a stronger impact on the percolation of the MCC. It is interesting to notice here that, since the relative performance of the algorithms considered clearly depends on both inter-layer degree correlations and edge overlap, there is no algorithm that clearly outperforms all the others. This is made evident in Fig. 6.13, where we highlight the behaviour of the ranking of the six heuristics

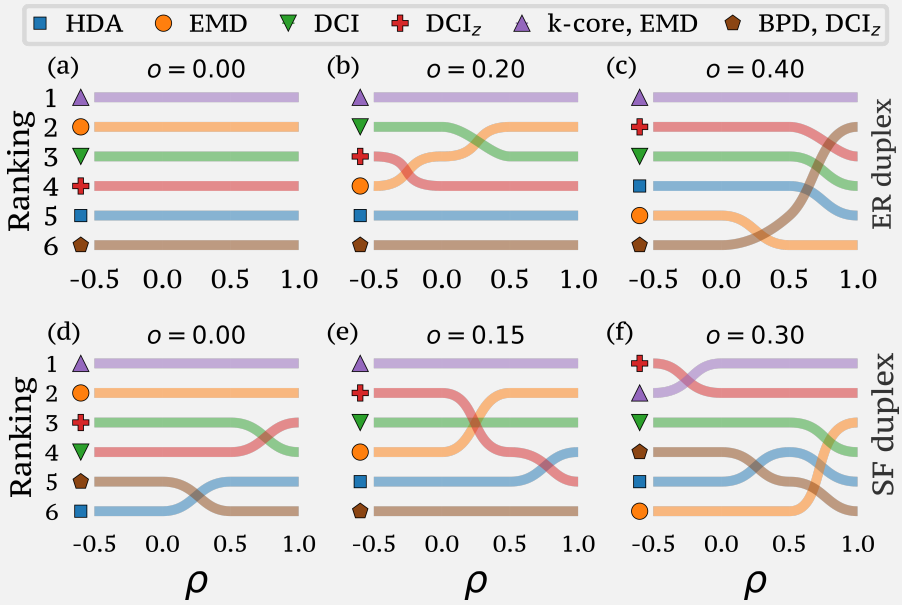

Figure 6.13. ***Effect of overlap and inter-layer degree correlation on the targeted strategies in synthetic duplex networks***. *We report the rankings of the six targeted attack strategies presented in Fig. 6.12 for different values of edge overlap and different network topology, namely, Erdös-Rényi (a-c) and scale–free (d-f), as a function of four values of inter-layer degree correlations. Interestingly, the (k-core, EMD) Pareto-efficient strategy has a considerably better performance in most of the conditions considered. By contrast, the EMD strategy appears to have a good performance only in duplex networks with negligible or small overlap.*

based on increasing size of the critical set q (i.e., the algorithm ranked first is the one providing the smallest critical set). Although the (k-core, EMD) Pareto-efficient strategy seems to perform consistently well across the board, being ranked first or second more often that the other five strategies, there are several combinations of layer structure, edge overlap, and inter-layer degree correlations for which other algorithms perform much better.

6.3.4 Optimal percolation in real-world multiplex networks

One of the main aims behind the study of targeted attacks is to try to find efficient ways to mitigate the fragility of real-world infrastructures, which are normally characterised by layer heterogeneity, non-negligible edge overlap, and inter-layer degree correlations. For this reason, we tested the targeted attack strategies presented above in 26 real-world multiplex networks [60].

The size of the critical set found by each of the algorithms is reported in Table 6.1. The systems considered in the Table range in size from few dozens to thousands of nodes, with different values of edge overlap and inter-layer degree correlations. Since many of those multiplex networks have more than two layers, for each system we considered the duplex sub-networks corresponding to the pairs of layers yielding the largest MCC, as already presented for instance in the main text of Ref. [371].

Unsurprisingly, there is no single strategy that works better than all the others in all the cases. What is surprising instead is that those strategies yielding the best performance when considering synthetic duplex systems, e.g., the (k-core, EMD) Pareto-efficient algorithm, do not perform as well in real-world systems. By contrast, DCI and DCI$_z$ quite often find the smallest critical set. This can be easily visualised in Fig. 6.14(a), where we plot the relative amount of times (i.e. performance rate) that a certain strategy identifies the smallest critical set in all the 26 real-world duplex networks considered. The best-performing strategy here is DCI , with a rate of 42%, followed by DCI$_z$ (35%) and EMD (31%). We also considered the pair performance, that is defined as the relative number of times that at least one of two algorithms identifies the smallest critical set. The results are reported in Fig. 6.14(b). Remarkably, combinations of targeted attacks including DCI and DCI$_z$ yield the best pair performance rate, where the pair [DCI , EMD] is able to find the smallest critical set in 58% of the cases. Overall, these results warn against the quest to find a single targeted attack strategy that performs well whatever the multiplex network it is applied to. In particular, the generalisation to real-world networks of targeting strategies that perform well in specific classes of synthetic graphs can result in the gross overestimation of the robustness of a system.

Data set	MCC	ρ	o_{norm}	HDA	EMD	DCI	DCI_z	k-core, EMD	k-core, DCI	k-core, DCI_z	CoreHD, EMD	CoreHD, DCI	CoreHD, DCI_z	$CI_{\ell=2}$, EMD	$CI_{\ell=2}$, DCI	$CI_{\ell=2}$, DCI_z	BPD, EMD	BPD, DCI	BPD, DCI_z	CoreHD, CoreHD	$CI_{\ell=2}$, $CI_{\ell=2}$	BPD, BPD
Air. FR-U2	28	0.25	0.14	4	5	4	4	4	4	4	4	4	4	4	4	4	5	4	5	4	4	5
Air. AA-DL	191	0.72	0.60	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	9	8	9
Air. AA-UA	204	0.77	0.58	9	7	7	7	8	7	7	8	8	8	9	8	9	9	8	8	9	9	8
UK Train L 26-41	59	0.38	0.19	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5
UK Train L 30-41	43	0.19	0.18	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
ArXiv L 2-6	916	0.85	0.77	91	119	106	106	139	108	108	115	101	100	101	104	103	137	121	115	117	104	119
ArXiv L 3-6	790	0.92	0.81	85	84	98	98	115	103	103	96	99	99	85	97	97	106	100	96	106	87	107
CS Aarhus L 1-5	58	0.32	0.63	14	15	15	12	13	15	15	12	15	15	12	14	14	15	15	13	18	14	14
FAO L 3-24	193	0.94	0.70	83	85	87	85	86	90	90	82	88	88	82	82	82	82	86	88	88	82	85
IMDb Com.-Dra.	181	0.97	0.82	88	90	87	89	88	90	90	91	90	90	88	90	89	90	89	89	89	88	89
Terr. L Tru.-Op.	61	0.22	0.50	18	16	15	15	19	19	19	17	17	17	19	19	19	15	15	15	17	21	21
Arabid. L 1-2	442	0.65	0.40	34	32	29	30	34	31	31	31	31	31	27	31	31	35	39	34	33	35	21
Drosoph. L 1-2	299	0.18	0.07	10	9	14	12	10	14	14	11	8	12	9	8	10	8	10	10	10	12	9
Drosoph. L 1-3	202	0.27	0.06	4	4	3	3	5	4	4	3	3	3	3	3	3	3	4	4	3	3	3
Drosoph. L 1-4	1024	0.13	0.09	26	26	27	26	35	39	39	25	37	31	24	27	30	38	42	38	40	33	42
Drosoph. L 2-3	449	0.68	0.35	49	48	50	53	52	50	50	47	48	49	55	51	51	53	54	51	54	54	52
Homo L 1-2	9312	0.57	0.32	1052	1058	1026	1021	1017	1045	1045	1042	1051	1069	1088	1060	1095	1088	1060	1092	1087	1101	1172
Homo L 1-5	3886	0.31	0.16	312	299	271	297	301	315	315	307	305	319	344	332	347	329	347	358	341	363	432
Homo L 2-5	4944	0.44	0.17	420	366	406	387	389	413	413	386	410	395	412	434	422	410	439	435	481	473	534
Hum. HIV L 1-2	144	0.54	0.41	4	4	3	3	4	3	3	4	2	2	4	4	4	4	3	3	2	4	3
Mus L 1-3	1059	0.56	0.37	60	60	52	50	57	52	52	61	69	65	58	57	60	61	68	56	60	58	63
S. Cerev. L 1-2	4531	0.36	0.10	785	757	768	769	743	785	785	795	782	787	853	823	831	811	811	825	859	885	956
S. Cerev. L 1-7	4720	0.28	0.07	982	940	898	905	911	930	930	994	958	942	1108	1000	1047	1025	953	973	1009	1025	1115
S. Pombe L 3-4	1112	0.20	0.14	56	41	50	52	54	59	59	44	54	52	54	54	53	57	62	60	72	79	75
S. Pombe L 3-6	956	0.14	0.06	39	32	40	35	42	38	38	35	40	35	38	38	37	45	41	39	50	49	58
S. Pombe L 4-6	2292	0.61	0.01	370	361	353	360	358	366	366	377	369	366	385	371	376	383	377	372	382	380	404

Table 6.1. ***Size of the attack sets identified by several targeted attack strategies for 26 different real-world duplex networks [2, 60].*** *For each data set we report the size of the initial MCC, the value of inter-layer degree correlations within the MCC (ρ), and the normalised edge overlap ($o_{\text{norm}} = o/o_{\max}$ where $o_{\max}$ is the maximum overlap in the corresponding configuration model ensemble [2, 41])[a]. The performance of each strategy heavily depends on the presence of edge overlap and inter-layer degree correlations. Interestingly, DCI , DCI_z , and Pareto-efficient strategies based on them perform better than all the other strategies. For methods that rely on random tie-breaking, the numbers reported correspond to the minimum value found by the method over 10 independent realisations. The size of the minimal attack set found on each duplex is underlined.*

[a] Since the maximum value $o_{\max}$ of edge overlap for a generic pair of layers is not known a-priori, in our simulations we computed an approximation of $o_{\max}$ by iteratively increasing the overlap of the system until no further increase is attainable using the model described in Chapter 5.2.1 (i.e. the termination criteria is such that the edge overlap does not increase after 5×10^7 random rewirings).

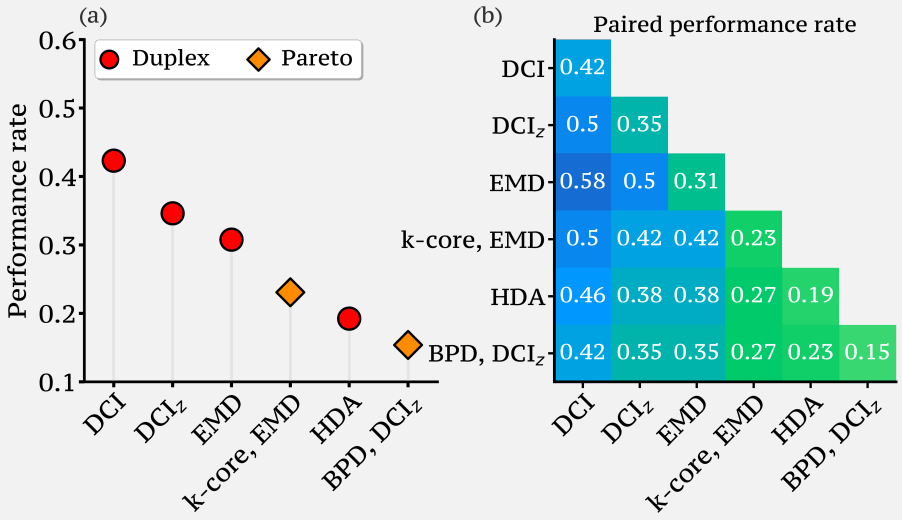

Figure 6.14. ***Optimal percolation in real-world systems***. *(a) Relative performance of the six targeted attack strategies for real-world duplex networks. The strategies that perform well in synthetic duplex networks are not the best ones when it comes to real-world systems. In panel (b) we show the overall pair performance rate, defined as the relative number of times either of a pair of algorithms identifies the smallest critical set. In this case, the combination of DCI and DCI$_z$ with other strategies results in the best performance rates. However, the fact that the highest value of pair performance is 58% suggests that we are still far from having fully understood the robustness of real-world systems to targeted attacks. Results are averaged over 10 realisations.*

6.4 Additional details on DCI and DCI$_z$

6.4.1 Dependence of Duplex Collective Influence score on edge overlap

Here we investigate the behaviour of the DCI score of a node i as a function of k_i^{int}, that is the degree of node i in the intersection graph, obtained by considering all and only the links that exist on both layers. Notice that k_i^{int} is intimately connected to the edge overlap around node i. Indeed, the fraction of edges attached to node i that exist in both layers can be expressed as $o_i = k_i^{\text{int}}/k_i^{\text{aggr}}$. Since we have $k_i^{\text{aggr}} = k_i^{[1]} + k_i^{[2]} - k_i^{\text{int}}$, the DCI score of node i can be rewritten as:

$$DCI(i) = \frac{k_i^{[1]} k_i^{[2]} - k_i^{\text{int}}}{k_i^{[1]} + k_i^{[2]} - k_i^{\text{int}}} \left[\sum_j a_{ij}^{[1]} (k_j^{[2]} - 1) + a_{ij}^{[2]} (k_j^{[1]} - 1) \right].$$

The term inside square brackets does not depend on k_i^{int}, so that we can just focus on the ratio outside, which can be conveniently rewritten as:

$$\frac{a - k_i^{\text{int}}}{b - k_i^{\text{int}}} \tag{6.6}$$

where we have set $a = k_i^{[1]} k_i^{[2]}$ and $b = k_i^{[1]} + k_i^{[2]}$. Notice that $k_i^{\text{int}} \in \left[0, \min\left(k_i^{[1]}, k_i^{[2]}\right)\right]$ and in particular $k_i^{\text{int}} = 0$ if the neighbourhoods of node i at the two layers are disjoint, while $k_i^{\text{int}} = \min\left(k_i^{[1]}, k_i^{[2]}\right)$ if the intersection between those two neighbourhoods is maximal, where the case $k_i^{[1]} = k_i^{[2]}$ corresponds to identical neighbourhoods on the two layers. It is easy to show that Eq. (6.6) is an increasing function of k_i^{int} for $a > b$, which holds whenever $\min\left(k_i^{[1]}, k_i^{[2]}\right) > 1$. This means that, all other things being equal, a node having degree larger than one on both layers will have a larger DCI score if it has a larger edge overlap. A similar reasoning holds for DCI$_z$.

6.4.2 DCI in multiplex networks with identical layers

It is easy to show that in a duplex network with identical layers the ranking of nodes induced by the DCI score defined in Eq. (6.3) coincides with that induced by the CI score on the corresponding aggregated graph with $\ell = 1$. In fact, if the two layers are identical, we have $a_{ij}^{[1]} = a_{ij}^{[2]} = a_{ij}\, \forall i, j = 1, \ldots, N$ and also $k_i^{[1]} = k_i^{[2]} = k_i^{\text{int}} = k_i^{\text{aggr}} = k_i\, \forall i = 1, \ldots, N$, so we get $DCI(i) = 2(k_i - 1) \sum_j a_{ij}(k_j - 1) = 2CI_{\ell=1}(i)$, which means that the two rankings are identical.

6.4.3 Time complexity

6.4.3.1 Time complexity of DCI and DCI$_z$

The adaptive targeted strategies based on DCI and DCI$_z$ require to re-compute the DCI scores of all the remaining nodes after each node is removed. An implementation with simple data

structures (basically, the list of neighbours of each node) guarantees a worst-case time complexity $\mathcal{O}(N^2 \log N)$, where N is the number of nodes of the graph. Indeed, the initial DCI (or DCI$_z$) score of all the nodes can be computed in $\mathcal{O}(K)$ (where K is the total number of edges of the multiplex), and sorted in $\mathcal{O}(N \log N)$. The removal of the i-th node from the network will modify the DCI scores of all its neighbours on the two layers, which are at most $N - i - 1$. Since we need to keep the list of DCI scores ordered, the usage of simple structures requires to sort again the scores, which has time complexity $\mathcal{O}(N \log N)$ at each step. As a consequence, updating DCI scores throughout the percolation procedure has time complexity $\mathcal{O}(N^2 \log N)$. A direct computation and update of the size of the LMCC would run in (N^3), but its efficiency can be improved to $\mathcal{O}(N^{1.2})$ by using the algorithm explained in Refs. [373, 374]. So overall the DCI (DCI$_z$) algorithm has time complexity $\mathcal{O}(N^2 \log N)$. However, the usage of a max-heap to store and update the list of DCI scores would guarantee a worst-case time complexity of $\mathcal{O}(N^{1.2} + K \log N)$, which is dominated by $\mathcal{O}(N^{1.2})$ in sparse graphs.

6.4.3.2 Time complexity of Pareto-efficient strategies

The time complexity of Pareto-efficient strategies can be expressed as $\mathcal{O}(F + S)$, where $\mathcal{O}(S)$ is the time complexity of computing and updating the scores used for multi-objective optimisation, while $\mathcal{O}(F)$ is the time complexity of computing and updating the Pareto-front throughout the percolation procedure. As mentioned in Chapter 4, identifying the Pareto Front at each step has time complexity $\mathcal{O}(N \log N)$ when the number of objective functions m is at most $m = 3$, which is the case for all the Pareto-efficient strategies considered in Chapter 6.3. If the number of functions to optimise is $m > 3$, then the time complexity becomes $\mathcal{O}(N(\log N)^{m-2})$ [279]. As a consequence $\mathcal{O}(F) = \mathcal{O}(N^2 \log N)$ in the worst case. The time complexity of computing and updating the scores depends on the details of the functions used, but all the functions we used are dominated by $\mathcal{O}(N^2 \log N)$.

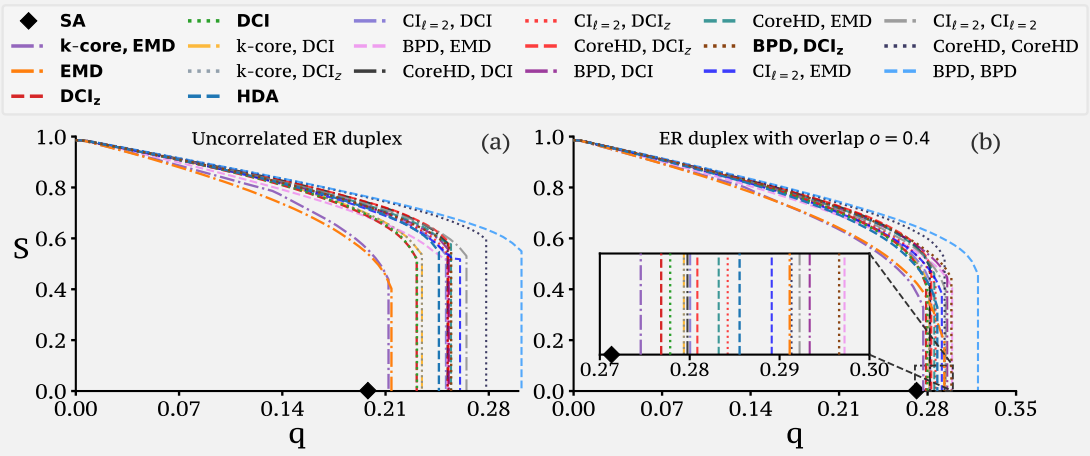

Figure 6.15. ***Performance of targeted attack strategies in synthetic duplex systems.*** *We report the percolation diagram (relative size of the LMCC vs q) for 20 different attack strategies for the same duplex networks respectively reported in Fig. 6.10(a) and in Fig. 6.10(c) of Chapter 6.3. Notice that Pareto-efficient strategies combining only single-layer metrics do not perform well in the two cases reported. By contrast, methods which effectively combine single and multi-layer information yield the best performance. Labels are sorted in ascending order of size of the critical set from panel (a). We highlighted in bold the six strategies presented in Fig. 6.10 of Chapter 6.3. Results averaged over 20 realisations.*

6.5 Additional results on synthetic networks

Here, we report the results obtained by the multiplex targeted strategies constructed by considering all Pareto-efficient combinations of different of single and multi-layer methods. In Fig. 6.15, we show the percolation diagrams (including also the results already presented in Fig. 6.10) in duplex networks with no overlap (panel a) and with high overlap (panel b). Clearly, strategies that incorporate multilayer information (i.e., HDA, EMD, DCI , and DCI$_z$), as well as Pareto-efficient strategy that take one of them into account, perform consistently better than those based exclusively on single-layer metrics [i.e., $(CI_{\ell=2}, CI_{\ell=2})$, (CoreHD, CoreHD), and (BPD, BPD)]. This is because, as noted in Refs. [366, 367, 375] and in Chapter 6.3, the presence of interdependencies in the multiplex structure deeply affects the overall robustness of duplex networks against random and targeted attacks, and this information is not present in either of the layers considered separately.

In Fig. 6.16 we report the size of the critical attack set q as a function of structural edge

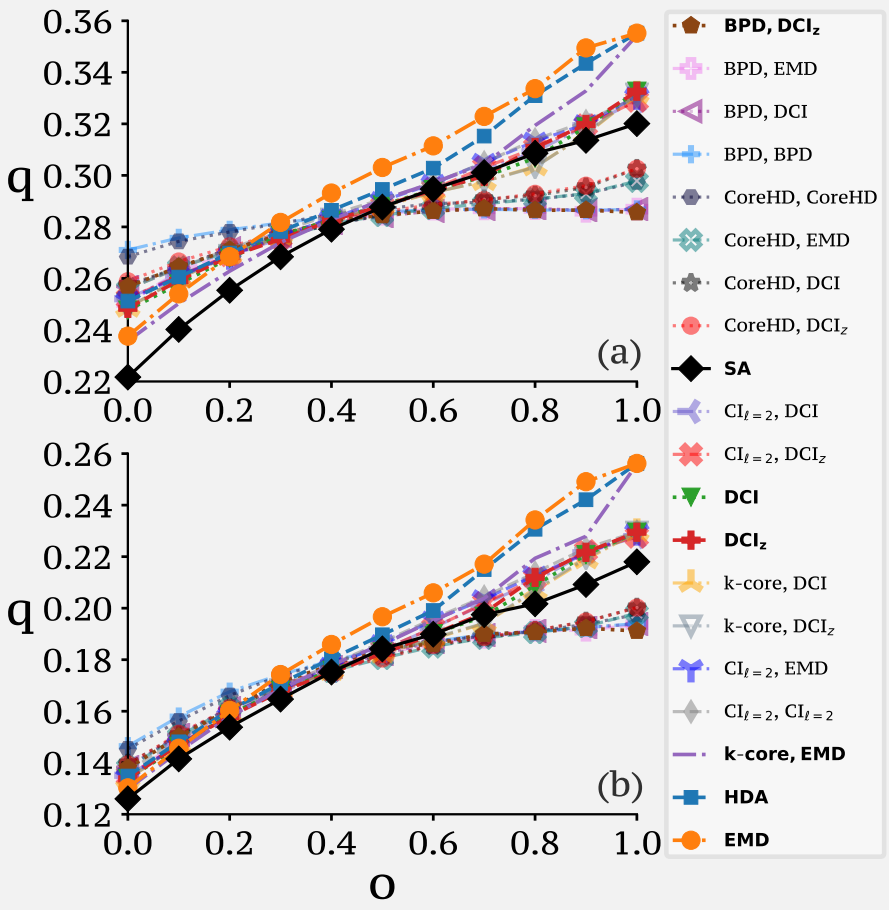

Figure 6.16. ***Performance of targeted attack strategies in synthetic duplex networks with tunable overlap.*** *We report the size of the critical attack set q as a function of edge overlap for the 20 different attack strategies in the same duplex networks as in Fig. 6.11 of Chapter 6.3. Labels are sorted in ascending order of size of the critical set from panel (a) when $o = 1$ [i.e., it is analogous to the single-layer percolation]. Results averaged over 20 realisations. Error bars are smaller than the marker size.*

overlap. As expected, the best performing targeted strategies for $o = 1$ are those based on BPD, which is the best-performing strategy on single-layer graphs [364, 365]. It is interesting to notice that some Pareto-efficient strategies outperform Simulated Annealing for large values of overlap (same implementation as the one presented in Ref. [367] with temperature steps equal to 10^{-7}).

Finally, in Fig. 6.17 we show the size of the critical set found by each of the 20 strategies for different combinations of interlayer degree correlations and edge overlap (same conditions presented in Fig. 6.12). It is clear that Pareto-efficient strategies combining multi- and single-

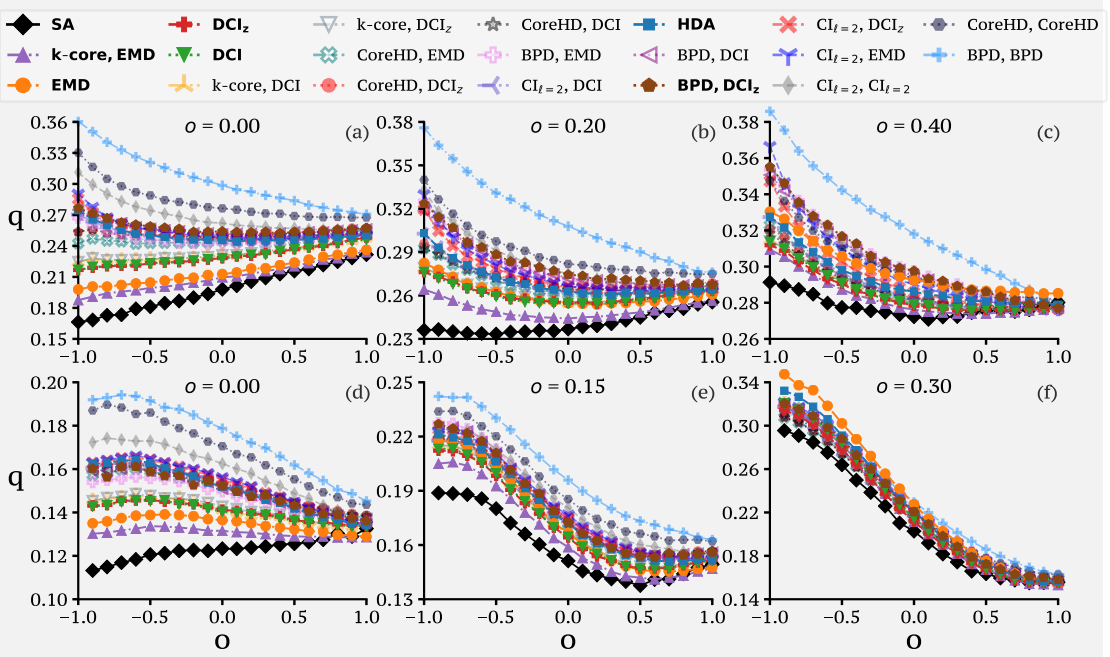

Figure 6.17. ***Performance of targeted attacks in synthetic duplex systems with overlap and inter-layer degree correlation.*** *We report the size of the critical set q found by each of the 20 attack strategies as a function of the inter-layer degree correlations coefficient for the same duplex networks as in Fig. 6.12. Notice that also in this case the attack strategies which combine only single-layer metrics perform quite poorly compared to the other Pareto-efficient methods. Labels are sorted in ascending order of q when considering $\rho = -1$ of panel (a), while in bold we highlighted the methods presented in Fig. 6.12. Results averaged over 20 realisations. Error bars are smaller than the marker size.*

layer information perform better than the others also in this case, and especially much better than methods relying only on single-layer metrics. This is even more evident when the duplex has $o \approx 0$ while the gap becomes smaller as the overlap increases, as expected.

Conclusion

In the last twenty years, network science has completely shaken the scientific horizon of many fields, including biology, physics, engineering, social science, sociology, ecology, computer science, epidemiology, and neuroscience. One of the main reasons behind its success is the intrinsic flexibility and adaptability that the theoretical and numerical tools have been shown to provide. Networks can effectively capture the intricate structure of interactions of many complex systems and hence provide a robust mathematical framework to study them. For example, network science paved the way for the systematic characterisation of human and animal societies, the spreading of epidemics and misinformation, but it also provided fascinating tools to understand how epileptic seizures and arrhythmias occur. In a quest to develop increasingly refined models to characterise real-world systems, researchers have gradually added new levels of complexity into the modelling, for instance, by including direction, weights, sign, or time within the edges of the network representations. Many of these efforts contributed to trigger three promising and prolific lines of research, where higher-order network models play a central role [238][1]. In this thesis, we focused on one of these three lines of research: the multilayer framework, in which the interactions of a system's components are decomposed into links of different types. On the one hand, a multiplex network generally combines significantly more information than any of its layers or compressed single-layer representations. On the other hand, more data are not always a bliss,

[1] These tree research directions involve (i) multilayer higher-order models with multiple link types [36, 37], (ii) combinatorial higher-order models, such as simplicial complexes [239], and (iii) non-Markovian higher-order network models [376]

even though the current trends in data science seem to suggest otherwise. Not all additional data available are indeed informative, and quite often more data implies more redundancy and more noise. As a result, one of the crucial challenges in the data science era consists of constructing new methods aimed at filtering out eventual noise and redundancies from the raw data in order to speed up possible analysis. In this regard, we took a step towards the study and modelling of real-world multiplex systems by employing tools of information and multi-objective optimisation theory.

After providing a broad overview on the impact of information theory on both single- and multilayer systems in Chapters 1-2, in the first part of this thesis we examined in detail the problem of finding lower-dimensional representations of a multiplex network that can exhibit the same structural and dynamical richness as the full multilayer graph. To address this problem, we proposed an algorithmic information-theoretic approach to evaluate the complexity of multiplex networks, defined as the ratio between the Kolmogorov complexity of the multiplex and that of the corresponding aggregated graph. In this way, we quantitatively assessed to which extent a given multiplex representation of a system is more informative than a single-layer graph. We then employed the same measure to detect redundancy in a multiplex network and to obtain meaningful lower-dimensional representations of a system. We finally demonstrated that such method allows to retain most of the structural complexity of the original system as well as the salient characteristics determining the behaviour of dynamical processes happening on it (e.g. the preservation of the epidemic threshold in reduced multiplex networks). We released an open-source library implementing our approach in Ref. [377].

In the second part of the thesis, instead, we shifted the focus to Pareto optimisation principles, which naturally arise from the concept of multi-objective optimisation theory. After a general introduction on the topic in Chapter 4, we linked for the first time multi-objective optimisation theory with the multilayer network representation. In particular, we stressed that an effective combination of multi-objective optimisation techniques and tools of multiplex network

theory often provide novel insights on both the modelling and analysis of real-world multiplex networks. On the one side, these approaches provided reasonable explanations for the evolution of real-world transportation systems. In particular, we showed that it is possible to characterise the technological advancement of a continent and the effectiveness of the network of individual carriers, throughout the systematic exploration of the feasible local improvements in the efficiency-competition plane at the level of single service providers. These insights provide valuable information regarding the placement of new routes or when comparing alternative expansion strategies. On the other side, we demonstrated that Pareto principles provide an extremely advantageous framework to generalise optimal single-layer percolation strategies to the case of multiplex networks. This is made possible by combining structural descriptors computed on different layers with genuinely multiplex information in a Pareto-efficient space, where each objective function represents a generic structural node descriptor. We presented an extensive comparison between several state-of-the-art heuristics on a set of synthetic and real-world multiplex systems showing that, in most of the cases, Pareto-efficient targeted strategies consistently outperformed the simple generalisations proposed in the past years in the context of synthetic networks. In addition, we have shown that the generalisations of the collective influence method [362] for duplex networks outperform all the other existing methods when applied to real-world duplex systems. We made available all the codes used for these analyses in Ref. [378].

At the end of this journey, this thesis provides the first algorithmic information-theoretic method to assess when a multiplex network representation really matters and, at the same time, paves the way for the construction of a general framework, rooted in optimisation principles, to characterise several real-world multiplex networks. But, more importantly, some of the methods and principles presented in this work can be regarded as a solid springboard for future applications in temporal networks, networks with metadata, and systems framed in terms of high-order interactions, such as simplicial complexes and hypergraphs [239].

Appendices

Appendix A

Appendix of Chapter 3

A.1 Numerical approaches for node interdependence

The encoding of a multiplex network through the prime-weight matrix Ω allows to define an efficient algorithm for the computation of node interdependence. As introduced in Chapter 2.3, the node interdependence is defined as follows:

$$\lambda_i = \frac{1}{N-1} \sum_{\substack{j \in N \\ j \neq i}} \frac{\psi_{ij}}{\sigma_{ij}}$$

where ψ_{ij} represents the number of shortest paths from i to j that use edges lying in at least two layers, while σ_{ij} represents the total number of shortest paths between node i and j in the multiplex. With a naive approach, the computation of such quantity requires computing the number of shortest paths in the multiplex, which usually scales exponentially with the number of layers. If we consider the prime-weight matrix Ω associated with a multiplex $\mathcal{M}$, we can compute the node interdependence of each node using elementary properties of composite numbers. The procedure is detailed below:

Algorithm: node interdependence λ_i

1. Calculate the unweighted aggregate W and prime-weight matrix $\Omega = \{\Omega_{ij}\}$ associated with the multiplex $\mathcal{M}$.

Enumerate all the shortest paths from node i using the Breadth-First Search (BFS) algorithm [65] applied to the graph associated with W. Set $\lambda_i \leftarrow 0$.

2. Set $\sigma_{ij} \leftarrow 0$ and $\psi_{ij} \leftarrow 0$.

3. We indicate the generic shortest path from i to j as the sequence $T = \{i, n_1, n_2, \ldots, n_k, j\}$, where $j \neq i$. We also indicate as $f(\Omega_{ij})$ the number of prime factors of Ω_{ij} .

4. Compute $P = \prod_{s=i, s\in T}^{n_k} f(\Omega_{s,\,s+1})$ and set $\sigma_{ij} \leftarrow \sigma_{ij} + P$.

5. Set $G \leftarrow GCD(\Omega_{i,\,n_1}, \Omega_{n_1,\,n_2}, \ldots, \Omega_{n_k,\,j})$. If G is not equal to 1, then $\psi_{ij} \leftarrow \psi_{ij} + f(G)$. Repeat from step 3. for all the shortest paths from i to j.

6. Set $\lambda_i \leftarrow \lambda_i + \frac{\sigma_{ij} - \psi_{ij}}{\sigma_{ij}}$. Repeat from step 2. for all the nodes j different from i.

In the algorithm, $GCD(a, b)$ is the greatest common divisor of a and b (i.e., the largest positive integer that divides both a and b without a remainder). The algorithm combines two major ingredients from number theory, namely the unique factorisation theorem (i.e. using the prime-weight matrix) and the properties of GCD. On the one hand, with the prime-weight matrix it is possible to count all the available shortest paths between two pairs of nodes in the multiplex networks. On the other hand, the GCD properties lead to a fast distinction between paths, selecting paths lying in just one layer to the others.

A.2 Multiplex data sets

The data sets introduced in Chapter 3.4 for the reducibility comparisons are: (i) the undirected routes of the 11 lines of the Barcelona tube network (https://www.tmb.cat/), (ii) the 9 lines of the Berlin tube (https://www.berlin.de/en/public-transportation/), (iii) the 17 lines of the Beijing subway (https://www.bjsubway.com/), and (iv) the scientific collaboration among countries (APS countries) obtained considering the papers published in the journals of the American Physical Society. For the latter, starting with the multiplex data set introduced in [224], we constructed a weighted multiplex collaboration network, in which nodes represent countries and a link connects two countries if scientists based in those countries co-authored a paper together. Authors having multiple affiliations were considered as belonging to multiple countries. The weight on each link represents the number of co-authorship relations between the corresponding two countries. In our analysis the unweighted version of such system has been used. All these data sets are accessible in Ref. [377]

In addition, the time-varying data sets used in Chapter 3.5 are: (i) the IMDb co-starring network [224], (ii) the financial multiplex network constructed from price time series of 35 major assets in NYSE and NASDAQ [379], (iii) the physics collaboration multiplex network of the American Physical Society (APS) and Web of Science (WOS) [224], and (iv) the FAO food import/export multiplex network (http://www.fao.org/statistics/databases). From each original data set (i,iii,iv), we constructed a time varying multiplex network by partitioning the original system in temporal windows of one year. In this process, we associate to each time window the corresponding static multiplex network containing all the links registered in that year.

Bibliography

[1] A. Santoro, V. Latora, G. Nicosia, and V. Nicosia, *Pareto optimality in multilayer network growth*, Phys. Rev. Lett. **121**, 128302 (2018).

[2] A. Santoro and V. Nicosia, *Algorithmic complexity of multiplex networks*, Phys. Rev. X **10**, 021069 (2020).

[3] A. Santoro and V. Nicosia, *Optimal percolation in correlated multilayer networks with overlap*, Phys. Rev. Research **2**, 033122 (2020).

[4] A. Patanè, A. Santoro, P. Conca, G. Carapezza, A. La Magna, V. Romano, and G. Nicosia, *Multi-objective optimization and analysis for the design space exploration of analog circuits and solar cells*, Engineering Applications of Artificial Intelligence **62**, 373 (2017).

[5] A. Patanè, A. Santoro, V. Romano, A. La Magna, and G. Nicosia, *Enhancing quantum efficiency of thin-film silicon solar cells by pareto optimality*, Journal of Global Optimization **72**, 491 (2018).

[6] J. C. W. Billings, M. Hu, G. Lerda, A. N. Medvedev, F. Mottes, A. Onicas, A. Santoro, and G. Petri, *Simplex2vec embeddings for community detection in simplicial complexes*, arXiv preprint arXiv:1906.09068 (2019).

[7] A. Bassolas, A. Santoro, S. Sousa, S. Rognone, and V. Nicosia, *Optimising the mitigation of epidemic spreading through targeted adoption of contact tracing apps*, arXiv preprint arXiv:2102.13013 (2021).

[8] E. Bullmore and O. Sporns, *The economy of brain network organization*, Nature Reviews Neuroscience **13**, 336 (2012).

[9] D. A. Drachman, Do we have brain to spare? (2005).

[10] S. Herculano-Houzel, *The human brain in numbers: a linearly scaled-up primate brain*, Frontiers in human neuroscience **3**, 31 (2009).

[11] V. Latora, V. Nicosia, and G. Russo, *Complex Networks: Principles, Methods and Applications* (Cambridge University Press, 2017).

[12] S. Boccaletti, V. Latora, Y. Moreno, M. Chavez, and D.-U. Hwang, *Complex networks: Structure and dynamics*, Physics reports **424**, 175 (2006).

[13] A.-L. Barabási *et al.*, *Network science* (Cambridge University Press, 2016).

[14] A. Barrat, M. Barthelemy, R. Pastor-Satorras, and A. Vespignani, *The architecture of complex weighted networks*, Proc. Natl. Acad. Sci. U.S.A. **101**, 3747 (2004).

[15] M. Barthélemy, A. Barrat, R. Pastor-Satorras, and A. Vespignani, *Characterization and modeling of weighted networks*, Physica a: Statistical mechanics and its applications **346**, 34 (2005).

[16] P. Holme and J. Saramäki, *Temporal networks*, Physics reports **519**, 97 (2012).

[17] P. Holme and J. Saramäki, in *Temporal networks* (Springer, 2013) pp. 1–14.

[18] P. Holme and J. Saramäki, *Temporal Network Theory* (Springer, 2019).

[19] A. Barrat, M. Barthelemy, and A. Vespignani, *Dynamical Processes on Complex Networks* (Cambridge University Press, 2008).

[20] D. S. Bassett and O. Sporns, *Network neuroscience*, Nature neuroscience **20**, 353 (2017).

[21] R. Pastor-Satorras, C. Castellano, P. Van Mieghem, and A. Vespignani, *Epidemic processes in complex networks*, Rev. Mod. Phys. **87**, 925 (2015).

[22] E. Bullmore and O. Sporns, *Complex brain networks: graph theoretical analysis of structural and functional systems*, Nature reviews neuroscience **10**, 186 (2009).

[23] O. Sporns, *Networks of the Brain* (MIT press, 2010).

[24] D. Centola, *The spread of behavior in an online social network experiment*, science **329**, 1194 (2010).

[25] S. Horvath, B. Zhang, M. Carlson, K. V. Lu, S. Zhu, R. M. Felciano, M. F. Laurance, W. Zhao, S. Qi, Z. Chen, Y. Lee, A. C. Scheck, L. M. Liau, H. Wu, D. H. Geschwind, P. G. Febbo, H. I. Kornblum, T. F. Cloughesy, S. F. Nelson, and P. S. Mischel, *Analysis of oncogenic signaling networks in glioblastoma identifies aspm as a molecular target*, Proceedings of the National Academy of Sciences **103**, 17402 (2006).

[26] M. R. Carlson, B. Zhang, Z. Fang, P. S. Mischel, S. Horvath, and S. F. Nelson, *Gene connectivity, function, and sequence conservation: predictions from modular yeast co-expression networks*, BMC Genomics **7**, 40 (2006).

[27] S. Horvath, *Weighted network analysis: applications in genomics and systems biology* (Springer Science & Business Media, 2011).

[28] V. Colizza, A. Barrat, M. Barthélemy, and A. Vespignani, *The role of the airline transportation network in the prediction and predictability of global epidemics*, Proceedings of the National Academy of Sciences **103**, 2015 (2006).

[29] A. P. y Piontti, N. Perra, L. Rossi, N. Samay, and A. Vespignani, *Charting the Next Pandemic: Modeling Infectious Disease Spreading in the Data Science Age* (Springer, 2018).

[30] Q. Zhang, K. Sun, M. Chinazzi, A. P. y Piontti, N. E. Dean, D. P. Rojas, S. Merler, D. Mistry, P. Poletti, L. Rossi, *et al.*, *Spread of zika virus in the americas*, Proceedings of the National Academy of Sciences **114**, E4334 (2017).

[31] A. Vespignani, H. Tian, C. Dye, J. O. Lloyd-Smith, R. M. Eggo, M. Shrestha, S. V. Scarpino, B. Gutierrez, M. U. Kraemer, J. Wu, *et al.*, *Modelling covid-19*, Nature Reviews Physics , 1 (2020).

[32] M. Chinazzi, J. T. Davis, M. Ajelli, C. Gioannini, M. Litvinova, S. Merler, A. P. y Piontti, K. Mu, L. Rossi, K. Sun, *et al.*, *The effect of travel restrictions on the spread of the 2019 novel coronavirus (covid-19) outbreak*, Science **368**, 395 (2020).

[33] M. U. Kraemer, C.-H. Yang, B. Gutierrez, C.-H. Wu, B. Klein, D. M. Pigott, L. Du Plessis, N. R. Faria, R. Li, W. P. Hanage, *et al.*, *The effect of human mobility and control measures on the covid-19 epidemic in china*, Science **368**, 493 (2020).

[34] B. F. Maier and D. Brockmann, *Effective containment explains subexponential growth in recent confirmed covid-19 cases in china*, Science **368**, 742 (2020).

[35] M. Gilbert, G. Pullano, F. Pinotti, E. Valdano, C. Poletto, P.-Y. Boëlle, E. d'Ortenzio, Y. Yazdanpanah, S. P. Eholie, M. Altmann, *et al.*, *Preparedness and vulnerability of african countries against importations of covid-19: a modelling study*, The Lancet **395**, 871 (2020).

[36] S. Boccaletti, G. Bianconi, R. Criado, C. I. Del Genio, J. Gómez-Gardenes, M. Romance, I. Sendina-Nadal, Z. Wang, and M. Zanin, *The structure and dynamics of multilayer networks*, Phys. Rep. **544**, 1 (2014).

[37] M. Kivelä, A. Arenas, M. Barthelemy, J. P. Gleeson, Y. Moreno, and M. A. Porter, *Multilayer networks*, J. Complex Netw. **2**, 203 (2014).

[38] G. Bianconi, *Multilayer networks: structure and function* (Oxford University Press, 2018).

[39] C. D. Brummitt, G. Barnett, and R. M. D'Souza, *Coupled catastrophes: sudden shifts cascade and hop among interdependent systems*, Journal of The Royal Society Interface **12**, 10.1098/rsif.2015.0712 (2015).

[40] S. Gomez, A. Diaz-Guilera, J. Gomez-Gardenes, C. J. Perez-Vicente, Y. Moreno, and A. Arenas, *Diffusion dynamics on multiplex networks*, Phys. Rev. Lett. **110**, 028701 (2013).

[41] M. Diakonova, V. Nicosia, V. Latora, and M. San Miguel, *Irreducibility of multilayer network dynamics: the case of the voter model*, New J. Phys. **18**, 023010 (2016).

[42] V. Nicosia, P. S. Skardal, A. Arenas, and V. Latora, *Collective phenomena emerging from the interactions between dynamical processes in multiplex networks*, Phys. Rev. Lett. **118**, 138302 (2017).

[43] F. Battiston, V. Nicosia, V. Latora, and M. San Miguel, *Layered social influence promotes multiculturality in the axelrod model*, Scientific Reports **7** (2017).

[44] J. Sanz, C.-Y. Xia, S. Meloni, and Y. Moreno, *Dynamics of interacting diseases*, Physical Review X **4**, 041005 (2014).

[45] J. P. Gleeson, K. P. O'Sullivan, R. A. Baños, and Y. Moreno, *Effects of network structure, competition and memory time on social spreading phenomena*, Physical Review X **6**, 021019 (2016).

[46] A. Cook, H. A. Blom, F. Lillo, R. N. Mantegna, S. Micciché, D. Rivas, R. Vázquez, and M. Zanin, *Applying complexity science to air traffic management*, Journal of Air Transport Management **42**, 149 (2015).

[47] M. Barigozzi, G. Fagiolo, and D. Garlaschelli, *Multinetwork of international trade: A commodity-specific analysis*, Physical Review E **81**, 046104 (2010).

[48] S. Poledna, J. L. Molina-Borboa, S. Martínez-Jaramillo, M. Van Der Leij, and S. Thurner, *The multi-layer network nature of systemic risk and its implications for the costs of financial crises*, Journal of Financial Stability **20**, 70 (2015).

[49] L. Bargigli, G. Di Iasio, L. Infante, F. Lillo, and F. Pierobon, *The multiplex structure of interbank networks*, Quantitative Finance **15**, 673 (2015).

[50] M. Montagna and C. Kok, *Multi-layered interbank model for assessing systemic risk*, ECB Working Paper No. 1944. (2016).

[51] L. Cantini, E. Medico, S. Fortunato, and M. Caselle, *Detection of gene communities in multi-networks reveals cancer drivers*, Scientific reports **5**, 17386 (2015).

[52] L. Bennett, A. Kittas, G. Muirhead, L. G. Papageorgiou, and S. Tsoka, *Detection of composite communities in multiplex biological networks*, Scientific reports **5**, 10345 (2015).

[53] S. D. Reis, Y. Hu, A. Babino, J. S. Andrade Jr, S. Canals, M. Sigman, and H. A. Makse, *Avoiding catastrophic failure in correlated networks of networks*, Nature Physics **10**, 762 (2014).

[54] M. De Domenico, S. Sasai, and A. Arenas, *Mapping multiplex hubs in human functional brain networks*, Frontiers in Neuroscience **10**, 326 (2016).

[55] L. Cai, X. Wei, J. Liu, L. Zhu, J. Wang, B. Deng, H. Yu, and R. Wang, *Functional integration and segregation in multiplex brain networks for alzheimer's disease*, Frontiers in Neuroscience **14** (2020).

[56] A. Cardillo, J. Gómez-Gardeñes, M. Zanin, M. Romance, D. Papo, F. Pozo, and S. Boccaletti, *Emergence of network features from multiplexity*, Sci. Rep. **3**, 1344 (2013).

[57] R. Gallotti and M. Barthelemy, *Anatomy and efficiency of urban multimodal mobility*, Sci. Rep. **4**, 1 (2014).

[58] M. Szell, R. Lambiotte, and S. Thurner, *Multirelational organization of large-scale social networks in an online world*, Proc. Natl. Acad. Sci. U.S.A. **107**, 13636 (2010).

[59] P. J. Mucha, T. Richardson, K. Macon, M. A. Porter, and J.-P. Onnela, *Community structure in time-dependent, multiscale, and multiplex networks*, Science **328**, 876 (2010).

[60] M. De Domenico, V. Nicosia, A. Arenas, and V. Latora, *Structural reducibility of multilayer networks*, Nat. Comm. **6**, 1 (2015).

[61] M. Barthelemy, *The structure and dynamics of cities* (Cambridge University Press, 2016).

[62] P. Erdös and A. Rényi, *On random graphs i*, Publicationes Mathematicae Debrecen **6**, 290 (1959).

[63] D. J. Watts and S. H. Strogatz, *Collective dynamics of 'small-world'networks*, Nature **393**, 440 (1998).

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

[65] M. Newman, *Networks* (Oxford University Press, 2018).

[66] D. d. S. Price, *A general theory of bibliometric and other cumulative advantage processes*, Journal of the American society for Information science **27**, 292 (1976).

[67] R. Ferrer i Cancho and R. V. Solé, in *Statistical mechanics of complex networks* (Springer, 2003) pp. 114–126.

[68] M. T. Gastner and M. E. Newman, *The spatial structure of networks*, The European Physical Journal B-Condensed Matter and Complex Systems **49**, 247 (2006).

[69] M. Barthélemy and A. Flammini, *Optimal traffic networks*, Journal of Statistical Mechanics: Theory and Experiment **2006**, L07002 (2006).

[70] M. Barthélemy and A. Flammini, *Modeling urban street patterns*, Phys. Rev. Lett. **100**, 138702 (2008).

[71] R. Louf, P. Jensen, and M. Barthelemy, *Emergence of hierarchy in cost-driven growth of spatial networks*, Proceedings of the National Academy of Sciences **110**, 8824 (2013), http://www.pnas.org/content/110/22/8824.full.pdf .

[72] A. Fabrikant, E. Koutsoupias, and C. H. Papadimitriou, in *International Colloquium on Automata, Languages, and Programming* (Springer, 2002) pp. 110–122.

[73] R. M. D'Souza, C. Borgs, J. T. Chayes, N. Berger, and R. D. Kleinberg, *Emergence of tempered preferential attachment from optimization*, Proc Natl Acad Sci U S A **104**, 6112 (2007).

[74] S. Valverde, R. F. Cancho, and R. V. Solé, *Scale-free networks from optimal design*, EPL (Europhysics Letters) **60**, 512 (2002).

[75] F. Papadopoulos, M. Kitsak, M. Á. Serrano, M. Boguñá, and D. Krioukov, *Popularity versus similarity in growing networks*, Nature **489**, 537 (2012).

[76] A.-L. Barabási, *Luck or reason*, Nature **489**, 507 (2012).

[77] V. Cutello, G. Narzisi, and G. Nicosia, *A multi-objective evolutionary approach to the protein structure prediction problem*, Journal of The Royal Society Interface **3**, 139 (2006), http://rsif.royalsocietypublishing.org/content/3/6/139.full.pdf .

[78] V. Nicosia, P. E. Vértes, W. R. Schafer, V. Latora, and E. T. Bullmore, *Phase transition in the economically modeled growth of a cellular nervous system*, Proceedings of the National Academy of Sciences **110**, 7880 (2013), http://www.pnas.org/content/110/19/7880.full.pdf .

[79] L. F. Seoane and R. Solé, *Phase transitions in pareto optimal complex networks*, Phys. Rev. E **92**, 032807 (2015).

[80] O. Shoval, H. Sheftel, G. Shinar, Y. Hart, O. Ramote, A. Mayo, E. Dekel, K. Kavanagh, and U. Alon, *Evolutionary trade-offs, pareto optimality, and the geometry of phenotype space*, Science **336**, 1157 (2012).

[81] H. Sheftel, O. Shoval, A. Mayo, and U. Alon, *The geometry of the pareto front in biological phenotype space*, Ecology and Evolution **3**, 1471 (2013).

[82] J. Goñi, A. Avena-Koenigsberger, N. Velez de Mendizabal, M. P. van den Heuvel, R. F. Betzel, and O. Sporns, *Exploring the morphospace of communication efficiency in complex networks*, PLOS ONE **8**, 1 (2013).

[83] A. Avena-Koenigsberger, J. Goñi, R. F. Betzel, M. P. van den Heuvel, A. Griffa, P. Hagmann, J.-P. Thiran, and O. Sporns, *Using pareto optimality to explore the topology and dynamics of the human connectome*, Philosophical Transactions of the Royal Society B: Biological Sciences **369**, 20130530 (2014).

[84] C. E. Shannon, *A mathematical theory of communication*, Bell System Technical Journal **27**, 379 (1948).

[85] T. M. Cover and J. A. Thomas, *Elements of information theory* (John Wiley & Sons, 2012).

[86] R. W. Yeung, *Information theory and network coding* (Springer Science & Business Media, 2008).

[87] G. J. Klir, *Uncertainty and information* (Wiley-interscience Hoboken (NJ), 2006).

[88] R. Ash, *Information Theory* (Dover Publications, 1990).

[89] I. Csiszar and J. Körner, *Information theory: coding theorems for discrete memoryless systems* (Cambridge University Press, 2011).

[90] F. M. Reza, *An introduction to information theory* (Courier Corporation, 1994).

[91] D. J. MacKay and D. J. Mac Kay, *Information theory, inference and learning algorithms* (Cambridge university press, 2003).

[92] W. McGill, *Multivariate information transmission*, Transactions of the IRE Professional Group on Information Theory **4**, 93 (1954).

[93] K. T. Hu, *On the amount of information*, Teoriya Veroyatnostei i ee Primeneniya **7**, 447 (1962).

[94] N. Abramson, *Information Theory and Coding* (McGraw-Hill, 1963).

[95] C. Finn and J. T. Lizier, *Generalised measures of multivariate information content*, Entropy **22**, 216 (2020).

[96] P. L. Williams and R. D. Beer, *Nonnegative decomposition of multivariate information*, arXiv preprint arXiv:1004.2515 (2010).

[97] N. Bertschinger, J. Rauh, E. Olbrich, and J. Jost, in *Proceedings of the European conference on complex systems 2012* (Springer, 2013) pp. 251–269.

[98] A. Kraskov, H. Stögbauer, R. G. Andrzejak, and P. Grassberger, *Hierarchical clustering using mutual information*, Europhys. Lett. **70**, 278 (2005).

[99] A. Lancichinetti, S. Fortunato, and J. Kertész, *Detecting the overlapping and hierarchical community structure in complex networks*, New J. Phys. **11**, 033015 (2009).

[100] P. A. Estévez, M. Tesmer, C. A. Perez, and J. M. Zurada, *Normalized mutual information feature selection*, IEEE Trans. Neural Netw. **20**, 189 (2009).

[101] J. R. Vergara and P. A. Estévez, *A review of feature selection methods based on mutual information*, Neural Comput. Appl. **24**, 175 (2014).

[102] S. Kullback and R. A. Leibler, *On information and sufficiency*, The annals of mathematical statistics **22**, 79 (1951).

[103] S. Kullback, *Information theory and statistics* (Courier Corporation, 1997).

[104] J. L. W. V. Jensen *et al.*, *Sur les fonctions convexes et les inégalités entre les valeurs moyennes*, Acta mathematica **30**, 175 (1906).

[105] W. Rudin *et al.*, *Principles of mathematical analysis*, Vol. 3 (McGraw-hill New York, 1964).

[106] K. Sayood, *Information theory and cognition: A review*, Entropy **20**, 706 (2018).

[107] L. Itti and P. Baldi, *Bayesian surprise attracts human attention*, Vision research **49**, 1295 (2009).

[108] P. Baldi and L. Itti, *Of bits and wows: A bayesian theory of surprise with applications to attention*, Neural Networks **23**, 649 (2010).

[109] P. S. Resnik, *Selection and information: a class-based approach to lexical relationships*, IRCS Technical Reports Series , 200 (1993).

[110] M. Light and W. Greiff, *Statistical models for the induction and use of selectional preferences*, Cognitive Science **26**, 269 (2002).

[111] R. Levy, *Expectation-based syntactic comprehension*, Cognition **106**, 1126 (2008).

[112] J. Murdock, C. Allen, and S. DeDeo, *Exploration and exploitation of victorian science in darwin's reading notebooks*, Cognition **159**, 117 (2017).

[113] A. T. Barron, J. Huang, R. L. Spang, and S. DeDeo, *Individuals, institutions, and innovation in the debates of the french revolution*, Proc. Natl. Acad. Sci. U.S.A. **115**, 4607 (2018).

[114] J. Lin, *Divergence measures based on the shannon entropy*, IEEE Trans. Inf. Theor. **37**, 145 (2006).

[115] O. Sporns, *Complexity*, Scholarpedia **2**, 1623 (2007).

[116] M. Mitchell, *Complexity: A guided tour* (Oxford University Press, 2009).

[117] S. Lloyd, *Measures of complexity: a nonexhaustive list*, IEEE Control Syst. Mag. **21**, 7 (2001).

[118] B. A. Huberman and T. Hogg, *Complexity and adaptation*, Phys. D **2**, 376–384 (1986).

[119] W. Weaver, *Science and complexity*, American Scientist **36** (1948).

[120] A. N. Kolmogorov, *Three approaches to the quantitative definition of information*, International journal of computer mathematics **2**, 157 (1968).

[121] G. J. Chaitin, *Algorithmic information theory*, IBM journal of research and development **21**, 350 (1977).

[122] A. Turing, *On computable numbers, with an application to the entscheidungsproblem*, Proceedings of the London mathematical society **2**, 230 (1937).

[123] C. E. Shannon, *A universal turing machine with two internal states*, Automata studies **34**, 157 (1956).

[124] M. Margenstern, *Frontier between decidability and undecidability: a survey*, Theoretical Computer Science **231**, 217 (2000).

[125] K. Gödel, *Über formal unentscheidbare sätze der principia mathematica und verwandter systeme i*, Monatshefte für mathematik und physik **38**, 173 (1931).

[126] D. Hilbert and W. Ackerman, *Theoretische Logik* (Springer, 1928).

[127] A. Church, *An unsolvable problem of elementary number theory*, American journal of mathematics **58**, 345 (1936).

[128] A. Church, *A note on the entscheidungsproblem*, The journal of symbolic logic **1**, 40 (1936).

[129] N. Cutland, *Computability: An introduction to recursive function theory* (Cambridge University Press, 1980).

[130] G. J. Chaitin, A. Arslanov, and C. Calude, *Program-size complexity computes the halting problem*, Tech. Rep. (Department of Computer Science, The University of Auckland, New Zealand, 1995).

[131] B. Mandelbrot, *How long is the coast of britain? statistical self-similarity and fractional dimension*, Science **156**, 636 (1967).

[132] B. B. Mandelbrot, *The fractal geometry of nature*, Vol. 173 (WH freeman New York, 1983).

[133] P. S. Addison, *Fractals and chaos: an illustrated course* (CRC Press, 1997).

[134] P. Grassberger, *Toward a quantitative theory of self-generated complexity*, International Journal of Theoretical Physics **25**, 907 (1986).

[135] M. Gell-Mann and S. Lloyd, *Information measures, effective complexity, and total information*, Complexity **2**, 44 (1996).

[136] J. P. Crutchfield and K. Young, *Inferring statistical complexity*, Phys. Rev. Lett. **63**, 105 (1989).

[137] C. H. Bennett, in *A Half-Century Survey on The Universal Turing Machine* (Oxford University Press, Inc., USA, 1988) p. 227–257.

[138] C. Adami and N. J. Cerf, *Physical complexity of symbolic sequences*, Physica D **137**, 62 (2000).

[139] C. Adami, *Sequence complexity in darwinian evolution*, Complexity **8**, 49 (2002).

[140] A. Kolmogorov, *Logical basis for information theory and probability theory*, IEEE Transactions on Information Theory **14**, 662 (1968).

[141] R. J. Solomonoff, *A formal theory of inductive inference. part i*, Information and control **7**, 1 (1964).

[142] R. J. Solomonoff, *A formal theory of inductive inference. part ii*, Information and control **7**, 224 (1964).

[143] G. J. Chaitin, *On the length of programs for computing finite binary sequences*, Journal of the ACM (JACM) **13**, 547 (1966).

[144] H. Zenil, N. A. Kiani, and J. Tegnér, *A review of graph and network complexity from an algorithmic information perspective*, Entropy **20**, 551 (2018).

[145] H. Zenil, N. A. Kiani, and J. Tegnér, *Low-algorithmic-complexity entropy-deceiving graphs*, Phys. Rev. E **96**, 012308 (2017).

[146] H. Zenil, N. A. Kiani, and J. Tegnér, in *Semin. Cell Dev. Biol.*, Vol. 51 (Elsevier, 2016) pp. 32–43.

[147] P. Grassberger, *Finite sample corrections to entropy and dimension estimates*, Phys. Lett. A **128**, 369 (1988).

[148] M. Li, P. Vitányi, *et al.*, *An introduction to Kolmogorov complexity and its applications*, Vol. 3 (Springer, 2008).

[149] G. J. Chaitin, *Information, randomness & incompleteness: papers on algorithmic information theory*, Vol. 8 (World Scientific, 1990).

[150] H. Zenil, *A review of methods for estimating algorithmic complexity: Options, challenges, and new directions*, Entropy **22**, 612 (2020).

[151] A. Shen, V. A. Uspensky, and N. Vereshchagin, *Kolmogorov complexity and algorithmic randomness*, Vol. 220 (American Mathematical Soc., 2017).

[152] J.-P. Delahaye and H. Zenil, *Numerical evaluation of algorithmic complexity for short strings: A glance into the innermost structure of randomness*, Applied Mathematics and Computation **219**, 63 (2012).

[153] F. Soler-Toscano, H. Zenil, J.-P. Delahaye, and N. Gauvrit, *Calculating kolmogorov complexity from the output frequency distributions of small turing machines*, PLOS ONE **9**, 1 (2014).

[154] R. J. Solomonoff, in *A preliminary report on a general theory of inductive inference* (United States Air Force, Office of Scientific Research, 1960).

[155] L. A. Levin, *Laws of information conservation (nongrowth) and aspects of the foundation of probability theory*, Problemy Peredachi Informatsii **10**, 30 (1974).

[156] C. S. Calude, *Information and randomness: an algorithmic perspective* (Springer Science & Business Media, 2013).

[157] N. Rashevsky, *Life, information theory, and topology*, The bulletin of mathematical biophysics **17**, 229 (1955).

[158] E. Trucco, *A note on the information content of graphs*, Bulletin of Mathematical Biology **18**, 129 (1956).

[159] A. Mowshowitz, *Entropy and the complexity of graphs: I. an index of the relative complexity of a graph*, The bulletin of mathematical biophysics **30**, 175 (1968).

[160] M. Dehmer and A. Mowshowitz, *A history of graph entropy measures*, Information Sciences **181**, 57 (2011).

[161] M. Dehmer, *Information processing in complex networks: Graph entropy and information functionals*, Applied Mathematics and Computation **201**, 82 (2008).

[162] F. Passerini and S. Severini, *Quantifying complexity in networks: the von neumann entropy*, International Journal of Agent Technologies and Systems (IJATS) **1**, 58 (2009).

[163] E. Estrada and N. Hatano, *Statistical-mechanical approach to subgraph centrality in complex networks*, Chemical Physics Letters **439**, 247 (2007).

[164] R. V. Solé and S. Valverde, in *Complex networks* (Springer, 2004) pp. 189–207.

[165] R. V. Solé, *Phase Transitions* (Princeton University Press, 2011).

[166] M. De Domenico and J. Biamonte, *Spectral entropies as information-theoretic tools for complex network comparison*, Physical Review X **6**, 041062 (2016).

[167] T. A. Schieber, L. Carpi, A. Díaz-Guilera, P. M. Pardalos, C. Masoller, and M. G. Ravetti, *Quantification of network structural dissimilarities*, Nature communications **8**, 13928 (2017).

[168] D. Chen, D.-D. Shi, M. Qin, S.-M. Xu, and G.-J. Pan, *Complex network comparison based on communicability sequence entropy*, Physical Review E **98**, 012319 (2018).

[169] J. P. Bagrow and E. M. Bollt, *An information-theoretic, all-scales approach to comparing networks*, Applied Network Science **4**, 45 (2019).

[170] E. Estrada and N. Hatano, *Communicability in complex networks*, Physical Review E **77**, 036111 (2008).

[171] S. Vishveshwara, K. Brinda, and N. Kannan, *Protein structure: insights from graph theory*, J. Theor. Comput. Chem. **1**, 187 (2002).

[172] G. Cimini, T. Squartini, F. Saracco, D. Garlaschelli, A. Gabrielli, and G. Caldarelli, *The statistical physics of real-world networks*, Nature Reviews Physics **1**, 58 (2019).

[173] J. C. Maxwell, *On the dynamical theory of gases*, Philosophical transactions of the Royal Society of London (1867).

[174] J. W. Gibbs, *Elementary principles in statistical mechanics: developed with especial reference to the rational foundation of thermodynamics* (C. Scribner's sons, 1902).

[175] J. Uffink, Compendium of the foundations of classical statistical physics (2006).

[176] F. Radicchi, D. Krioukov, H. Hartle, and G. Bianconi, *Classical information theory of networks*, Journal of Physics: Complexity **1**, 025001 (2020).

[177] E. T. Jaynes, *Information theory and statistical mechanics*, Physical review **106**, 620 (1957).

[178] E. T. Jaynes, *On the rationale of maximum-entropy methods*, Proceedings of the IEEE **70**, 939 (1982).

[179] P. W. Holland and S. Leinhardt, *An exponential family of probability distributions for directed graphs*, Journal of the american Statistical association **76**, 33 (1981).

[180] D. Strauss, *On a general class of models for interaction*, SIAM review **28**, 513 (1986).

[181] J. Park and M. E. Newman, *Statistical mechanics of networks*, Physical Review E **70**, 066117 (2004).

[182] M. E. Newman, S. H. Strogatz, and D. J. Watts, *Random graphs with arbitrary degree distributions and their applications*, Physical review E **64**, 026118 (2001).

[183] K. Anand and G. Bianconi, *Entropy measures for networks: Toward an information theory of complex topologies*, Physical Review E **80**, 045102 (2009).

[184] K. Anand and G. Bianconi, *Gibbs entropy of network ensembles by cavity methods*, Physical Review E **82**, 011116 (2010).

[185] T. Squartini, J. de Mol, F. den Hollander, and D. Garlaschelli, *Breaking of ensemble equivalence in networks*, Physical review letters **115**, 268701 (2015).

[186] T. Squartini and D. Garlaschelli, *Reconnecting statistical physics and combinatorics beyond ensemble equivalence*, arXiv preprint arXiv:1710.11422 (2017).

[187] G. Bianconi, *The entropy of randomized network ensembles*, EPL (Europhysics Letters) **81**, 28005 (2007).

[188] K. Anand, G. Bianconi, and S. Severini, *Shannon and von neumann entropy of random networks with heterogeneous expected degree*, Physical Review E **83**, 036109 (2011).

[189] G. Bianconi, *Entropy of network ensembles*, Phys. Rev. E **79**, 036114 (2009).

[190] R. Albert and A.-L. Barabási, *Statistical mechanics of complex networks*, Rev. Mod. Phys. **74**, 47 (2002).

[191] M. Rosvall, A. Trusina, P. Minnhagen, and K. Sneppen, *Networks and cities: An information perspective*, Physical Review Letters **94**, 028701 (2005).

[192] K. Sneppen, A. Trusina, and M. Rosvall, *Hide-and-seek on complex networks*, EPL (Europhysics Letters) **69**, 853 (2005).

[193] A. Trusina, M. Rosvall, and K. Sneppen, *Communication boundaries in networks*, Physical review letters **94**, 238701 (2005).

[194] M. Rosvall, P. Minnhagen, and K. Sneppen, *Navigating networks with limited information*, Physical Review E **71**, 066111 (2005).

[195] M. Rosvall, A. Grönlund, P. Minnhagen, and K. Sneppen, *Searchability of networks*, Physical Review E **72**, 046117 (2005).

[196] A. Fronczak and P. Fronczak, *Biased random walks in complex networks: The role of local navigation rules*, Physical Review E **80**, 016107 (2009).

[197] R. Sinatra, J. Gómez-Gardenes, R. Lambiotte, V. Nicosia, and V. Latora, *Maximal-entropy random walks in complex networks with limited information*, Physical Review E **83**, 030103 (2011).

[198] J. Goñi, M. P. van den Heuvel, A. Avena-Koenigsberger, N. V. de Mendizabal, R. F. Betzel, A. Griffa, P. Hagmann, B. Corominas-Murtra, J.-P. Thiran, and O. Sporns, *Resting-brain functional connectivity predicted by analytic measures of network communication*, Proc. Natl. Acad. Sci. U.S.A. **111**, 833 (2014).

[199] S. Fortunato, *Community detection in graphs*, Phys. Rep. **486**, 75 (2010).

[200] M. Rosvall and C. T. Bergstrom, *An information-theoretic framework for resolving community structure in complex networks*, Proc. Natl. Acad. Sci. U.S.A. **104**, 7327 (2007).

[201] M. Rosvall and C. T. Bergstrom, *Maps of random walks on complex networks reveal community structure*, Proc. Natl. Acad. Sci. U.S.A. **105**, 1118 (2008).

[202] M. Rosvall, D. Axelsson, and C. T. Bergstrom, *The map equation*, The European Physical Journal Special Topics **178**, 13 (2009).

[203] M. E. Newman, *Communities, modules and large-scale structure in networks*, Nature physics **8**, 25 (2012).

[204] S. Fortunato and D. Hric, *Community detection in networks: A user guide*, Physics reports **659**, 1 (2016).

[205] M. Girvan and M. E. Newman, *Community structure in social and biological networks*, Proceedings of the national academy of sciences **99**, 7821 (2002).

[206] M. E. Newman and M. Girvan, *Finding and evaluating community structure in networks*, Physical review E **69**, 026113 (2004).

[207] M. E. Newman, *Fast algorithm for detecting community structure in networks*, Physical review E **69**, 066133 (2004).

[208] A. Clauset, M. E. Newman, and C. Moore, *Finding community structure in very large networks*, Physical review E **70**, 066111 (2004).

[209] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, *Fast unfolding of communities in large networks*, Journal of statistical mechanics: theory and experiment **2008**, P10008 (2008).

[210] H. Zenil, F. Soler-Toscano, K. Dingle, and A. A. Louis, *Correlation of automorphism group size and topological properties with program-size complexity evaluations of graphs and complex networks*, Physica A: Statistical Mechanics and its Applications **404**, 341 (2014).

[211] G. Menichetti, D. Remondini, P. Panzarasa, R. J. Mondragón, and G. Bianconi, *Weighted multiplex networks*, PloS One **9** (2014).

[212] F. Battiston, V. Nicosia, and V. Latora, *The new challenges of multiplex networks: Measures and models*, Eur. Phys. J Spec. Top. **226**, 401 (2017).

[213] C. Granell, S. Gómez, and A. Arenas, *Dynamical interplay between awareness and epidemic spreading in multiplex networks*, Phys. Rev. Lett. **111**, 128701 (2013).

[214] E. Cozzo, R. A. Banos, S. Meloni, and Y. Moreno, *Contact-based social contagion in multiplex networks*, Phys. Rev. E **88**, 050801 (2013).

[215] H. Wang, Q. Li, G. D'Agostino, S. Havlin, H. E. Stanley, and P. Van Mieghem, *Effect of the interconnected network structure on the epidemic threshold*, Physical Review E **88**, 022801 (2013).

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

[217] Sociopatterns collaboration, `http://www.sociopatterns.org/`, accessed: Feb 2020.

[218] P. Sapiezynski, A. Stopczynski, D. D. Lassen, and S. Lehmann, *Interaction data from the copenhagen networks study*, Sci. Data **6**, 1 (2019).

[219] P. Holme, *Modern temporal network theory: a colloquium*, The European Physical Journal B **88**, 234 (2015).

[220] N. Masuda and R. Lambiotte, *A Guidance to Temporal Networks* (World Scientific, 2016).

[221] M. De Domenico, A. Solé-Ribalta, E. Cozzo, M. Kivelä, Y. Moreno, M. A. Porter, S. Gómez, and A. Arenas, *Mathematical formulation of multilayer networks*, Physical Review X **3**, 041022 (2013).

[222] R. Guimerá and L. A. N. Amaral, *Cartography of complex networks: modules and universal roles*, Journal of Statistical Mechanics: Theory and Experiment **2005**, P02001 (2005).

[223] R. Guimerá and L. A. N. Amaral, *Functional cartography of complex metabolic networks*, Nature **433**, 895 (2005).

[224] V. Nicosia and V. Latora, *Measuring and modeling correlations in multiplex networks*, Phys. Rev. E **92**, 032805 (2015).

[225] R. G. Morris and M. Barthelemy, *Transport on coupled spatial networks*, Phys. Rev. Lett. **109**, 128703 (2012).

[226] R. Criado, J. Flores, A. García del Amo, J. Gómez-Gardeñes, and M. Romance, *A mathematical model for networks with structures in the mesoscale*, International Journal of Computer Mathematics **89**, 291 (2012).

[227] L. Lacasa, V. Nicosia, and V. Latora, *Network structure of multivariate time series*, Sci. Rep. **5**, 15508 (2015).

[228] D. Cellai, E. López, J. Zhou, J. P. Gleeson, and G. Bianconi, *Percolation in multiplex networks with overlap*, Phys. Rev. E **88**, 052811 (2013).

[229] D. Cellai, S. N. Dorogovtsev, and G. Bianconi, *Message passing theory for percolation models on multiplex networks with link overlap*, Phys. Rev. E **94**, 032301 (2016).

[230] G. Bianconi, *Statistical mechanics of multiplex networks: Entropy and overlap*, Phys. Rev. E **87**, 062806 (2013).

[231] M. E. Newman, *Assortative mixing in networks*, Phys. Rev. Lett. **89**, 208701 (2002).

[232] M. E. Newman, *Mixing patterns in networks*, Phys. Rev. E **67**, 026126 (2003).

[233] R. Pastor-Satorras, A. Vázquez, and A. Vespignani, *Dynamical and correlation properties of the internet*, Phys. Rev. Lett. **87**, 258701 (2001).

[234] R. Pastor-Satorras and A. Vespignani, *Evolution and structure of the Internet: A statistical physics approach* (Cambridge University Press, 2007).

[235] S. Wasserman, K. Faust, *et al.*, *Social network analysis: Methods and applications*, Vol. 8 (Cambridge university press, 1994).

[236] V. Nicosia, G. Bianconi, V. Latora, and M. Barthelemy, *Growing multiplex networks*, Phys. Rev. Lett. **111**, 058701 (2013).

[237] V. Nicosia, G. Bianconi, V. Latora, and M. Barthelemy, *Nonlinear growth and condensation in multiplex networks*, Phys. Rev. E **90**, 042807 (2014).

[238] R. Lambiotte, M. Rosvall, and I. Scholtes, *From networks to optimal higher-order models of complex systems*, Nat. Phys. **15**, 313 (2019).

[239] F. Battiston, G. Cencetti, I. Iacopini, V. Latora, M. Lucas, A. Patania, J.-G. Young, and G. Petri, *Networks beyond pairwise interactions: structure and dynamics*, Physics Reports (2020).

[240] J. Iacovacci, Z. Wu, and G. Bianconi, *Mesoscopic structures reveal the network between the layers of multiplex data sets*, Phys. Rev. E **92**, 042806 (2015).

[241] T. P. Peixoto, *Entropy of stochastic blockmodel ensembles*, Phys. Rev. E **85**, 056122 (2012).

[242] G. Bianconi, P. Pin, and M. Marsili, *Assessing the relevance of node features for network structure*, Proc. Natl. Acad. Sci. U.S.A. **106**, 11433 (2009).

[243] APS website-PACS, `https://journals.aps.org/PACS`, accessed: Feb 2020.

[244] J. Iacovacci and G. Bianconi, *Extracting information from multiplex networks*, Chaos **26**, 065306 (2016).

[245] L. Danon, A. Diaz-Guilera, J. Duch, and A. Arenas, *Comparing community structure identification*, J. Stat. Mech. **2005**, P09008 (2005).

[246] F. Battiston, J. Iacovacci, V. Nicosia, G. Bianconi, and V. Latora, *Emergence of multiplex communities in collaboration networks*, PloS One **11** (2016).

[247] M. De Domenico, A. Lancichinetti, A. Arenas, and M. Rosvall, *Identifying modular flows on multilayer networks reveals highly overlapping organization in interconnected systems*, Phys. Rev. X **5**, 011027 (2015).

[248] A. Ben-Hur, A. Elisseeff, and I. Guyon, in *Biocomputing 2002* (World Scientific, 2001) pp. 6–17.

[249] W. Rand, *Objective criteria for the evaluation of clustering methods*, Journal of the American Statistical Association **66**, 846 (1971).

[250] L. Hubert and P. Arabie, *Comparing partitions*, Journal of classification **2**, 193 (1985).

[251] L. C. Carpi, T. A. Schieber, P. M. Pardalos, G. Marfany, C. Masoller, A. Díaz-Guilera, and M. G. Ravetti, *Assessing diversity in multiplex networks*, Sci. Rep. **9**, 1 (2019).

[252] N. Stanley, S. Shai, D. Taylor, and P. J. Mucha, *Clustering network layers with the strata multilayer stochastic block model*, IEEE Trans. Network Sci. Eng. **3**, 95 (2016).

[253] C. De Bacco, E. A. Power, D. B. Larremore, and C. Moore, *Community detection, link prediction, and layer interdependence in multilayer networks*, Phys. Rev. E **95**, 042317 (2017).

[254] T.-C. Kao and M. A. Porter, *Layer communities in multiplex networks*, J. Stat. Phys. **173**, 1286 (2018).

[255] A. Ghavasieh and M. De Domenico, *Enhancing transport properties in interconnected systems without altering their structure*, Phys. Rev. Research **2**, 013155 (2020).

[256] J. Briët and P. Harremoës, *Properties of classical and quantum jensen-shannon divergence*, Phys. Rev. A **79**, 052311 (2009).

[257] E. T. Bell, *Exponential numbers*, Am. Math. Monthly **41**, 411 (1934).

[258] Gzip compression protocol, `https://gzip.org`.

[259] F. Battiston, V. Nicosia, M. Chavez, and V. Latora, *Multilayer motif analysis of brain networks*, Chaos **27**, 047404 (2017).

[260] M. H. DeGroot and M. J. Schervish, *Probability and statistics* (Pearson Education, 2012).

[261] Bzip2 compression protocol, `https://bzip.org`.

[262] Lz4 compression protocol, `https://lz4.org`.

[263] J. H. Ward Jr, *Hierarchical grouping to optimize an objective function*, J. Am. Stat. Assoc. **58**, 236 (1963).

[264] F. Battiston, V. Nicosia, and V. Latora, *Efficient exploration of multiplex networks*, New J. Phys. **18**, 043035 (2016).

[265] K. Deb, *Multi-objective optimization using evolutionary algorithms*, Vol. 16 (John Wiley & Sons, 2001).

[266] L. M. Rios and N. V. Sahinidis, *Derivative-free optimization: a review of algorithms and comparison of software implementations*, Journal of Global Optimization **56**, 1247 (2013).

[267] K. Miettinen, *Nonlinear multiobjective optimization*, Vol. 12 (Springer Science & Business Media, 2012).

[268] S. S. Rao, *Engineering optimization: theory and practice* (John Wiley & Sons, 2019).

[269] D. P. Bertsekas, *Nonlinear programming*, Journal of the Operational Research Society **48**, 334 (1997).

[270] A. Patane, A. Santoro, J. Costanza, G. Carapezza, and G. Nicosia, *Pareto optimal design for synthetic biology*, IEEE transactions on biomedical circuits and systems **9**, 555 (2015).

[271] H. W. Kuhn and A. W. Tucker, in *Proceedings of the Second Berkeley Symposium on Mathematical Statistics and Probability* (University of California Press, Berkeley, Calif., 1951) pp. 481–492.

[272] A. M. Geoffrion, *Proper efficiency and the theory of vector maximization*, Journal of mathematical analysis and applications **22**, 618 (1968).

[273] G. Nicosia, S. Rinaudo, and E. Sciacca, in *International Conference on Innovative Techniques and Applications of Artificial Intelligence* (Springer, 2007) pp. 7–20.

[274] S. Bandaru and K. Deb, in *IEEE Congress on Evolutionary Computation* (2010) pp. 1–8.

[275] B. Tomoiagă, M. Chindriş, A. Sumper, A. Sudria-Andreu, and R. Villafafila-Robles, *Pareto optimal reconfiguration of power distribution systems using a genetic algorithm based on nsga-ii*, Energies **6**, 1439 (2013).

[276] T. Chen, M. Kwiatkowska, A. Simaitis, and C. Wiltsche, in *International Conference on Quantitative Evaluation of Systems* (Springer, 2013) pp. 322–337.

[277] M. Soleimani-Damaneh, *An optimization modelling for string selection in molecular biology using pareto optimality*, Applied Mathematical Modelling **35**, 3887 (2011).

[278] J. Branke, J. Branke, K. Deb, K. Miettinen, and R. Slowiński, *Multiobjective optimization: Interactive and evolutionary approaches*, Vol. 5252 (Springer Science & Business Media, 2008).

[279] H. T. Kung, F. Luccio, and F. P. Preparata, *On finding the maxima of a set of vectors*, J. ACM **22**, 469 (1975).

[280] M. Laumanns, L. Thiele, K. Deb, and E. Zitzler, *Combining convergence and diversity in evolutionary multiobjective optimization*, Evolutionary computation **10**, 263 (2002).

[281] M. Ehrgott, *Multicriteria optimization*, Vol. 491 (Springer Science & Business Media, 2005).

[282] K. Deb, in *Multiobjective Optimization* (Springer, 2008) pp. 59–96.

[283] J. L. Cohon, in *Design optimization* (Elsevier, 1985) pp. 163–191.

[284] C.-L. Hwang and A. S. M. Masud, *Multiple objective decision making—methods and applications: a state-of-the-art survey*, Vol. 164 (Springer Science & Business Media, 2012).

[285] M. Zeleny, *Multiple criteria decision*, Vol. 123 (Springer Science & Business Media, 2012).

[286] W. Rudin, *Real and complex analysis* (McGraw-Hill education, 2006).

[287] Y. Y. Haimes, S. L. Lasdon, and A. D. Wismer, *On a bicriterion formulation of the problems of integrated system identification and system optimization*, IEEE Transactions on Systems, Man, and Cybernetics **SMC-1**, 296 (1971).

[288] G.-H. Tzeng and J.-J. Huang, *Multiple attribute decision making: methods and applications* (CRC press, 2011).

[289] C.-L. Hwang, Y.-J. Lai, and T.-Y. Liu, *A new approach for multiple objective decision making*, Computers & operations research **20**, 889 (1993).

[290] M. Cheikh, B. Jarboui, T. Loukil, and P. Siarry, *A method for selecting pareto optimal solutions in multiobjective optimization*, Journal of Informatics and mathematical sciences **2**, 51 (2010).

[291] Z. Wang and G. P. Rangaiah, *Application and analysis of methods for selecting an optimal solution from the pareto-optimal front obtained by multiobjective optimization*, Industrial & Engineering Chemistry Research **56**, 560 (2017).

[292] D. E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*, 1st ed. (Addison-Wesley Longman Publishing Co., Inc., USA, 1989).

[293] K. Deb, A. R. Reddy, and G. Singh, *Optimal scheduling of casting sequence using genetic algorithms*, Materials and Manufacturing Processes **18**, 409 (2003).

[294] D. E. Goldberg and K. Deb, in *Foundations of genetic algorithms*, Vol. 1 (Elsevier, 1991) pp. 69–93.

[295] E. Zitzler, M. Laumanns, and L. Thiele, *Spea2: Improving the strength pareto evolutionary algorithm*, TIK-report **103** (2001).

[296] M. Reyes-Sierra, C. C. Coello, *et al.*, *Multi-objective particle swarm optimizers: A survey of the state-of-the-art*, International journal of computational intelligence research **2**, 287 (2006).

[297] A. J. Nebro, J. J. Durillo, J. Garcia-Nieto, C. C. Coello, F. Luna, and E. Alba, in *2009 IEEE Symposium on Computational Intelligence in Multi-Criteria Decision-Making (MCDM)* (IEEE, 2009) pp. 66–73.

[298] K. Deb and H. Jain, *An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part i: solving problems with box constraints*, IEEE transactions on evolutionary computation **18**, 577 (2013).

[299] A. Zhou, B.-Y. Qu, H. Li, S.-Z. Zhao, P. N. Suganthan, and Q. Zhang, *Multiobjective evolutionary algorithms: A survey of the state of the art*, Swarm and Evolutionary Computation **1**, 32 (2011).

[300] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, *A fast and elitist multiobjective genetic algorithm: Nsga-ii*, IEEE transactions on evolutionary computation **6**, 182 (2002).

[301] T. Okabe, Y. Jin, and B. Sendhoff, in *The 2003 Congress on Evolutionary Computation, 2003. CEC'03.*, Vol. 2 (IEEE, 2003) pp. 878–885.

[302] N. Riquelme, C. Von Lücken, and B. Baran, in *2015 Latin American Computing Conference (CLEI)* (IEEE, 2015) pp. 1–11.

[303] C. Audet, J. Bigeon, D. Cartier, S. Le Digabel, and L. Salomon, *Performance indicators in multiobjective optimization*, Optimization Online (2018).

[304] E. Zitzler and L. Thiele, in *International conference on parallel problem solving from nature* (Springer, 1998) pp. 292–301.

[305] M. Fleischer, in *International Conference on Evolutionary Multi-Criterion Optimization* (Springer, 2003) pp. 519–533.

[306] E. Zitzler, *Evolutionary algorithms for multiobjective optimization: Methods and applications*, Vol. 63 (Citeseer, 1999).

[307] H. Ishibuchi, R. Imada, Y. Setoguchi, and Y. Nojima, *How to specify a reference point in hypervolume calculation for fair performance comparison*, Evolutionary computation **26**, 411 (2018).

[308] D. Cellai and G. Bianconi, *Multiplex networks with heterogeneous activities of the nodes*, Physical Review E **93**, 032302 (2016).

[309] A. Halu, S. Mukherjee, and G. Bianconi, *Emergence of overlap in ensembles of spatial multiplexes and statistical mechanics of spatial interacting network ensembles*, Physical Review E **89**, 012806 (2014).

[310] T. P. Peixoto, *Inferring the mesoscale structure of layered, edge-valued, and time-varying networks*, Phys. Rev. E **92**, 042807 (2015).

[311] M. Bazzi, L. G. S. Jeub, A. Arenas, S. D. Howison, and M. A. Porter, *A framework for the construction of generative models for mesoscale structure in multilayer networks*, Phys. Rev. Research **2**, 023100 (2020).

[312] Mammult, https://github.com/KatolaZ/mammult.

[313] N. Momeni and B. Fotouhi, *Growing multiplex networks with arbitrary number of layers*, Phys. Rev. E **92**, 062812 (2015).

[314] V. Nicosia, G. Bianconi, V. Latora, and M. Barthelemy, *Nonlinear growth and condensation in multiplex networks*, Phys. Rev. E **90**, 042807 (2014).

[315] P. L. Krapivsky, S. Redner, and F. Leyvraz, *Connectivity of growing random networks*, Phys. Rev. Lett. **85**, 4629 (2000).

[316] R. Guimerá, S. Mossa, A. Turtschi, and L. N. Amaral, *The worldwide air transportation network: Anomalous centrality, community structure, and cities' global roles*, Proc. Natl. Acad. Sci. U.S.A. **102**, 7794 (2005).

[317] M. Zanin and F. Lillo, *Modelling the air transport with complex networks: A short review*, The European Physical Journal Special Topics **215**, 5 (2013).

[318] T. Verma, N. A. Araújo, and H. J. Herrmann, *Revealing the structure of the world airline network*, Sci. Rep. **4**, 1 (2014).

[319] A. Cook, *European air traffic management: principles, practice, and research* (Ashgate Publishing, Ltd., 2007).

[320] J. Tinbergen, *Shaping the world economy; suggestions for an international economic policy* (1962).

[321] P. Pöyhönen, *A tentative model for the volume of trade between countries*, Weltwirtschaftliches Archiv , 93 (1963).

[322] Multiplex Transportation data set, `http://github.com/andresantoro/tranet_data/` (2018).

[323] Multiplex data set, `http://www.maths.qmul.ac.uk/~vnicosia/sw.html` (2015).

[324] OpenFlights data set, `http://openflights.org/`, accessed: Nov 2017.

[325] Opentraintimes data set, `http://www.opentraintimes.com/`, accessed: Oct 2017.

[326] UK network rail data set, `http://www.networkrail.co.uk/`, accessed: Oct 2017.

[327] UK coaches data set, `http://www.travelinedata.org.uk/`, accessed: Oct 2017.

[328] W. K. Hastings, *Monte Carlo sampling methods using Markov chains and their applications*, Biometrika **57**, 97 (1970).

[329] T. W. Anderson, *On the distribution of the two-sample cramér-von mises criterion*, The Annals of Mathematical Statistics **33**, 1148 (1962).

[330] G. K. Zipf, *Human behavior and the principle of least effort: An introduction to human ecology* (Ravenio Books, 2016).

[331] M. Batty, *Rank clocks*, Nature **444**, 592 (2006).

[332] E. Zitzler, L. Thiele, M. Laumanns, C. M. Fonseca, and V. G. Da Fonseca, *Performance assessment of multiobjective optimizers: An analysis and review*, IEEE Transactions on evolutionary computation **7**, 117 (2003).

[333] GDP countries data set, `http://data.worldbank.org/indicator/NY.GDP.PCAP.CD`, accessed: Oct 2017.

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

[335] R. Cohen and S. Havlin, *Complex Networks: Structure, Robustness and Function* (Cambridge University Press, 2010).

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

[337] S.-W. Son, G. Bizhani, C. Christensen, P. Grassberger, and M. Paczuski, *Percolation theory on interdependent networks based on epidemic spreading*, EPL **97**, 16006 (2012).

[338] F. Radicchi and G. Bianconi, *Redundant interdependencies boost the robustness of multiplex networks*, Phys. Rev. X **7**, 011013 (2017).

[339] B. Min, S. D. Yi, K.-M. Lee, and K.-I. Goh, *Network robustness of multiplex networks with interlayer degree correlations*, Phys. Rev. E **89**, 042811 (2014).

[340] G. J. Baxter, G. Bianconi, R. A. da Costa, S. N. Dorogovtsev, and J. F. F. Mendes, *Correlated edge overlaps in multiplex networks*, Phys. Rev. E **94**, 012303 (2016).

[341] Y. Hu, D. Zhou, R. Zhang, Z. Han, C. Rozenblat, and S. Havlin, *Percolation of interdependent networks with intersimilarity*, Phys. Rev. E **88**, 052805 (2013).

[342] B. Min, S. Lee, K.-M. Lee, and K.-I. Goh, *Link overlap, viability, and mutual percolation in multiplex networks*, Chaos **72**, 49 (2015).

[343] G. T. Cantwell and M. E. J. Newman, *Message passing on networks with loops*, Proceedings of the National Academy of Sciences **116**, 23398 (2019).

[344] B. Karrer, M. E. J. Newman, and L. Zdeborová, *Percolation on sparse networks*, Phys. Rev. Lett. **113**, 208702 (2014).

[345] Y.-Y. Liu, J.-J. Slotine, and A.-L. Barabási, *Controllability of complex networks*, Nature **473**, 167 (2011).

[346] F. Altarelli, A. Braunstein, L. Dall'Asta, J. R. Wakeling, and R. Zecchina, *Containing epidemic outbreaks by message-passing techniques*, Phys. Rev. X **4**, 021024 (2014).

[347] K. ichiro Hashimoto, in *Automorphic Forms and Geometry of Arithmetic Varieties*, Advanced Studies in Pure Mathematics, Vol. 15, edited by K. Hashimoto and Y. Namikawa (Academic Press, 1989) pp. 211 – 280.

[348] J. Gao, S. V. Buldyrev, H. E. Stanley, and S. Havlin, *Networks formed from interdependent networks*, Nat. Phys. **8**, 40 (2012).

[349] F. Radicchi, *Percolation in real interdependent networks*, Nat. Phys. **11**, 597 (2015).

[350] G. Bianconi and F. Radicchi, *Percolation in real multiplex networks*, Phys. Rev. E **94**, 060301 (2016).

[351] F. Coghi, F. Radicchi, and G. Bianconi, *Controlling the uncertain response of real multiplex networks to random damage*, Phys. Rev. E **98**, 062317 (2018).

[352] R. Albert, H. Jeong, and A.-L. Barabási, *Error and attack tolerance of complex networks*, Nature **406**, 378 (2000).

[353] D. S. Callaway, M. E. J. Newman, S. H. Strogatz, and D. J. Watts, *Network robustness and fragility: Percolation on random graphs*, Phys. Rev. Lett. **85**, 5468 (2000).

[354] G. Dong, J. Gao, R. Du, L. Tian, H. E. Stanley, and S. Havlin, *Robustness of network of networks under targeted attack*, Phys. Rev. E **87**, 052804 (2013).

[355] D.-w. Zhao, L.-h. Wang, Y.-f. Zhi, J. Zhang, and Z. Wang, *The robustness of multiplex networks under layer node-based attack*, Sci. Rep. **6**, 24304 (2016).

[356] D. Kempe, J. Kleinberg, and É. Tardos, in *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining* (2003) pp. 137–146.

[357] M. Kitsak, L. K. Gallos, S. Havlin, F. Liljeros, L. Muchnik, H. E. Stanley, and H. A. Makse, *Identification of influential spreaders in complex networks*, Nat. Phys. **6**, 888 (2010).

[358] R. Pastor-Satorras and A. Vespignani, *Immunization of complex networks*, Phys. Rev. E **65**, 036104 (2002).

[359] R. Cohen, S. Havlin, and D. ben Avraham, *Efficient immunization strategies for computer networks and populations*, Phys. Rev. Lett. **91**, 247901 (2003).

[360] P. Clusella, P. Grassberger, F. J. Pérez-Reche, and A. Politi, *Immunization and targeted destruction of networks using explosive percolation*, Phys. Rev. Lett. **117**, 208301 (2016).

[361] A. Braunstein, L. Dall'Asta, G. Semerjian, and L. Zdeborová, *Network dismantling*, Proc. Natl. Acad. Sci. U.S.A. **113**, 12368 (2016).

[362] F. Morone and H. A. Makse, *Influence maximization in complex networks through optimal percolation*, Nature **524**, 65 (2015).

[363] S. Mugisha and H.-J. Zhou, *Identifying optimal targets of network attack by belief propagation*, Phys. Rev. E **94**, 012305 (2016).

[364] L. Zdeborová, P. Zhang, and H.-J. Zhou, *Fast and simple decycling and dismantling of networks*, Sci. Rep. **6**, 37954 (2016).

[365] S. Pei, J. Wang, F. Morone, and H. A. Makse, *Influencer identification in dynamical complex systems*, J. Complex Netw. , cnz029 (2019).

[366] S. Osat, A. Faqeeh, and F. Radicchi, *Optimal percolation on multiplex networks*, Nat. Comm. **8**, 1540 (2017).

[367] G. J. Baxter, G. Timár, and J. F. F. Mendes, *Targeted damage to interdependent networks*, Phys. Rev. E **98**, 032307 (2018).

[368] G. Bianconi and S. N. Dorogovtsev, *Percolation in networks of networks with random matching of nodes in different layers*, arXiv preprint arXiv:1411.4160 (2014).

[369] S. Brin and L. Page, *The anatomy of a large-scale hypertextual web search engine*, Computer networks and ISDN systems **30**, 107 (1998).

[370] F. Morone, B. Min, L. Bo, R. Mari, and H. A. Makse, *Collective influence algorithm to find influencers via optimal percolation in massively large social media*, Sci. Rep. **6**, 30062 (2016).

[371] K.-K. Kleineberg, L. Buzna, F. Papadopoulos, M. Boguñá, and M. A. Serrano, *Geometric correlations mitigate the extreme vulnerability of multiplex networks against targeted attacks*, Phys. Rev. Lett. **118**, 218301 (2017).

[372] Ş. Erkol, C. Castellano, and F. Radicchi, *Systematic comparison between methods for the detection of influential spreaders in complex networks*, Sci. Rep. **9**, 1 (2019).

[373] S. Hwang, S. Choi, D. Lee, and B. Kahng, *Efficient algorithm to compute mutually connected components in interdependent networks*, Phys. Rev. E **91**, 022814 (2015).

[374] D. Lee, S. Hwang, S. Choi, and B. Kahng, *Decremental dynamic algorithm to trace mutually connected clusters*, SoftwareX **7**, 273 (2018).

[375] S. V. Buldyrev, N. W. Shere, and G. A. Cwilich, *Interdependent networks with identical degrees of mutually dependent nodes*, Phys. Rev. E **83**, 016112 (2011).

[376] M. Rosvall, A. V. Esquivel, A. Lancichinetti, J. D. West, and R. Lambiotte, *Memory in network flows and its effects on spreading dynamics and community detection*, Nature communications **5**, 1 (2014).

[377] ALCOREM repository, `https://github.com/andresantoro/ALCOREM` (2020).

[378] Optimal multiplex percolation codes, `https://github.com/andresantoro/Multiplex_optimal_percolation/` (2020).

[379] N. Musmeci, V. Nicosia, T. Aste, T. Di Matteo, and V. Latora, *The multiplex dependency structure of financial markets*, Complexity **2017**, 9586064 (2017).