UNIVERSIDADE DE LISBOA
INSTITUTO SUPERIOR TÉCNICO

Complex networks analysis from an edge perspective

Andreia Sofia Monteiro Teixeira

Supervisor: Doctor Alexandre Paulo Lourenço Francisco
Co-Supervisor: Doctor Francisco João Duarte Cordeiro Correia dos Santos

Thesis approved in public session to obtain the PhD Degree in
Information Systems and Computer Engineering

Jury final classification: **Pass with Distinction**

2019

UNIVERSIDADE DE LISBOA
INSTITUTO SUPERIOR TÉCNICO

Complex networks analysis from an edge perspective

Andreia Sofia Monteiro Teixeira

Supervisor: Doctor Alexandre Paulo Lourenço Francisco
Co-Supervisor: Doctor Francisco João Duarte Cordeiro Correia dos Santos

Thesis approved in public session to obtain the PhD Degree in
Information Systems and Computer Engineering

Jury final classification: **Pass with Distinction**

Jury

Chairperson: Doctor José Manuel da Costa Alves Marques, Instituto Superior Técnico, Universidade de Lisboa

Members of the Committee:

Doctor Luís Jorge Mateus Silva Rocha, School of Informatics, Computer & Engineering, Indiana University, EUA;
Doctor Alexandre Paulo Lourenço Francisco, Instituto Superior Técnico, Universidade de Lisboa;
Doctor Pedro Manuel Pinto Ribeiro, Faculdade de Ciências, Universidade do Porto;
Doctor Luís Jorge Brás Monteiro Guerra e Silva, Instituto Superior Técnico, Universidade de Lisboa.

Funding Institutions
Universidade de Lisboa
Fundação para a Ciência e Tecnologia

2019

To have been always what I am – and so changed from what I was.

Samuel Becket

No fundo, todos temos necessidade de dizer quem somos
e o que é que estamos a fazer e a necessidade de deixar algo feito,
porque esta vida não é eterna
e deixar coisas feitas pode ser uma forma de eternidade.

José Saramago

Acknowledgments

More than a scientific journey, this PhD was a personal journey. I was fortunate to work in a fascinating field, full of challenges. These past four years have made me grow and made me more resilient. And although a PhD can make us feel like we are alone in the world, the challenges encountered were only possible to overcome thanks to very special people and institutions.

First of all, I would like to thank my supervisors, Alexandre Francisco and Francisco Santos, for their tireless and unconditional support. I could not have had better mentors. Their advice and experience made me a more proactive and perceptive student. They gave me the independence necessary to grow as a scientific researcher, never leaving my side when I needed.

I would also like to thank my co-authors, with whom I learn so much and without whom this path would not be so fulfilling: Pedro T. Monteiro, João Carriço, Mário Ramirez, Luis Russo, Fernando P. Santos, Pedro Souto and Francisco Fernandes.

To INESC-ID, a second home since my master's degree, my great gratitude for providing me with the environment I needed to be successful. In particular, I would like to thank Manuela Sado, Ana de Jesus and Vanda Fidalgo for the human support.

During my PhD I was a Teaching Assistant at the Department of Computer Science and Engineering, of Instituto Superior Técnico, Universidade de Lisboa, and since last September I have been an Invited Assistant Professor at the Department of Informatics, of Faculdade de Ciências, Universidade de Lisboa. I want to thank both institutions for providing me the opportunity of gaining teaching experience.

I am grateful to my friends, but especially to Eugénio Ribeiro, Nuno Capela, Jorge Oliveira, Isabel Saura and Ana Sofia Correia, who were fundamental in the most critical times. Also in the range of people who have helped to make the most difficult moments more bearable, I want to thank to Omnichord Records human frame. Being able to follow the creative process of such talented people was often therapeutic.

Obrigada à minha família. Por tudo. Este doutoramento é especialmente dedicado à minha mãe, Piedade Monteiro Teixeira, e ao meu pai, António Teixeira, os meus pilares. Portadores de uma força e perseverança inspiradoras, com o seu amor incondicional tornaram, sempre que possível, a minha vida mais fácil. Muito obrigada, de coração. Obrigada também ao meu irmão, Renato Teixeira, pelos seus

abraços que tanto transmitiram, e, claro, ao Bran, o meu terapeuta preferido.

This PhD would not have been possible without the doctoral fellowships of Universidade de Lisboa and Fundação para a Ciência e Tecnologia, among other funding support (Programa Incentivo/EEI/LA00-21/2014, EXCL/EEI-ESS/0257/2012, PTDC/EEI-SII/1937/2014, TUBITAK/0004/2014, SFRH/BD/12907-2/2017, and EU project BIRDS 690941-H2020-MSCA-RISE-2015). Thank you.

Andreia Sofia Monteiro Teixeira

Abstract

If we carefully observe our daily lives and the systems in which we participate, we naturally perceive that everything is somehow connected. From species evolution to social relations, acknowledging all the supply chain systems we depend on, networks portray the simplest representation of these systems. Notwithstanding this simplicity, these networks often underlie complex dynamics. Species and population evolution are subject to many complex interactions. Individuals' states – from individual choices, epidemic states, strategic behaviors, opinions, among others – are influenced by social ties and by the overall topology of interaction. These networks, called complex networks, show a prevalence of certain features, which are shared between completely different systems, thus defying the limits of the traditional techniques of analysis and intriguing the research community.

In this thesis we aim to contribute to the study of the relationship between structure and dynamics of these complex networks. Usually, the approaches to study complex networks are centered on the importance of nodes. However, it is our understanding that the edge-perspective analysis also provides fundamental and complementary information on the structure and behavior of complex networks. Given this, throughout this dissertation we approach complex networks under an edge perspective, centering our attention on the properties of the edges. In our contributions we provide new metrics, models, and computational tools. We start by contributing with a new edge centrality measure. Next, we focus on analyzing local patterns(or subgraphs) whose edges contain informative labels, highlighting that sometimes observing only nodes and edges, individually, is not enough to fully understand the dynamics and/or the structure of a system. Finally, we argue that representing a system with a single network is often insufficient to reproduce its behavior. Therefore, it is necessary to consider networks at multiple scales, i.e. networks of networks. In this context, we propose a new computational framework that allows us to model and simulate a system represented as a network of networks.

Keywords: Network Science; Complex Networks; Link Significance; Structural Balance; Large-Scale Simulations.

Resumo

Se observarmos com atenção o nosso quotidiano, e os sistemas em que participamos, é fácil observarmos que tudo está, de alguma forma, ligado. Da evolução de espécies às relações sociais, passando por todos os sistemas de fornecimento de bens, estas ligações pertencentes aos mais diversos domínios encontram nas redes de contacto (físico ou conceptual) a sua forma de representação mais simples. Não obstante essa simplicidade, as redes que representam estes sistemas têm como base dinâmicas complexas. Espécies e populações evoluem tendo em conta interações complexas, e os estados individuais – desde escolhas individuais, estados epidémicos, comportamentos estratégicos, opiniões, entre outros – são influenciados por laços sociais e pela topologia da rede de interação. Estas redes, denominadas por redes complexas, mostram uma prevalência de determinadas características, partilhadas entre sistemas completamente distintos, desafiando assim os limites das técnicas tradicionais de análise e intrigando a comunidade científica.

Nesta dissertação pretendemos contribuir para o estudo da relação entre estrutura e dinâmica destas redes complexas. Normalmente, a abordagem ao estudo de redes complexas é centrada na importância dos nós. No entanto, é do nosso entendimento que uma análise centrada nos arcos também fornece informação fundamental e complementar sobre a estrutura e o comportamento da rede. Consequentemente, ao longo desta dissertação abordaremos as redes complexas numa perspetiva centrada nas características dos arcos. As nossas contribuições contam com novas métricas, novos modelos e novas ferramentas computacionais. Começamos por contribuir com uma nova medida de centralidade para os arcos. De seguida, concentramo-nos em analisar padrões locais/sub-redes cujos arcos contêm etiquetas informativas, destacando que, por vezes, observar apenas nós e arcos, de forma individual, não é o suficiente para compreender completamente a dinâmica e/ou estrutura de um sistema. Por último, observamos que muitas vezes representar um sistema com apenas uma rede é insuficiente para reproduzir o seu comportamento, sendo necessário considerar redes com múltiplas escalas, i.e., redes de redes. Neste contexto, a nossa contribuição consiste numa nova ferramenta computacional que nos permite modelar e simular um sistema representado como uma rede de redes.

Palavras-Chave: Ciência das Redes; Redes Complexas; Significância de Arcos; Árvores de Extensão Mínima; Equilíbrio Estrutural; Redes de Redes; Simulações em Larga Escala.

Contents

List of Figures

List of Tables

List of Algorithms

Listings

1

Introduction

Contents

Everything is somehow connected. We have our friends and family with whom we relate with. We travel through roads to visit cities. We drink water from a water supply network and the electricity in our houses comes from an electrical power grid. Even our existence is based on a series of complex interactions between blood vessels and organs, between thousands of genes and transcription factors of other genes, and our brain is one of the most incredible networks where billions of neurons communicate through trillions of synapses. We could continue, almost infinitely, describing more complex systems as are those of economics, communications, collaborations, citations, just to mention a few more.

Graphs portray the simplest representation of complex systems. They can represent a wide variety of entities – represented by the nodes – which are related or interact with each other – through the edges/links. With the explosive growth of real networks and structured datasets, a new class of graphs came to light. This kind of graphs has some prevailing characteristics, exposing the limits of traditional graph theory techniques. The size of the graphs ranges from thousands to billions of vertices, making brute force approaches no longer feasible. The graphs are mostly sparse. The small world phenomenon is observed, i.e., graphs have small distances among vertices and reveal clustering effects. Finally, the vertex degree distribution usually satisfies a power law. The fact that many different networks share these characteristics has intrigued the research community and, in the beginning of this millennium, a new area of research on graphs has been rapidly developing - Network Science [15, 19, 27, 39, 54, 55, 131].

Network Science is the field that continuously provides theory and methods to study the relationship between the structure and function of these real networks. While conceptually simple, networks often underlie complex dynamics and reasoning about their behaviour is increasingly difficult. From mathematicians, physicists, social scientists, biologists to computer scientists, among many more, this interdisciplinary field has been gathering efforts of very different research communities. It has become crucial to cross information, models and approaches from different fields to better understand structure and dynamics of these complex networks.

Let us illustrate a little bit more. If we want to understand the evolution of species, we need both biologists and computer scientists to develop rigorous and efficient models and tools that allow us to process and understand the data that is becoming available from sequencing. If we want to understand how people interact with each other, and why/how they tend to adapt their relations among time, we may need several scientists from different areas, as psychologists or social scientists, to understand the data available and also to try to develop simulation models that can explain human behavior. A third example can be the need to follow and predict financial volatility, which demands efforts of researchers from economics and computational science to process and analyze market stock data.

In the last two decades, Network Science has provided a considerable amount of statistics, measures, tools and frameworks to understand the interplay between structure and dynamics of complex

systems. The aim of this thesis is to contribute to the Network Science field with more insights about both structure and dynamics in complex networks from an edge perspective. In the following sections we identify three different ways of looking into networks, all taking into account edge properties, and we describe our new valuable contributions.

1.1 Challenges in Network Science

Network Science is a relatively young research field, with plenty of interesting open research questions. To study a complex network, how it behaves and how structure and function are related, there are different possible approaches. For instance, although a network is a set of edges, the research community has been focusing on a node centered perspective. Indeed, the traditional approach is to analyze the degree distribution, identify which nodes have higher centrality and then apply a model based on both. Such node-centered way of looking into networks, identifying which nodes are more important given a criteria, has been extremely useful to identify and characterize network properties and also to develop a considerable amount of spreading models that take advantage of highly connected nodes. Nonetheless, despite receiving less direct attention, edges can also provide valuable insights about both the structure and dynamics of complex networks.

The strength of studying networks based on an edge perspective approach is that we can take advantage of the real meaning of the connections between nodes/entities. Even when looking to subgraphs or motifs, we can choose to look at them as a set of edges, analyzing the properties of the edges and how they influence the overall dynamics. Also, we must not forget that there are complex systems that are better translated as networks of networks, where the communication/interaction between the networks is relied on the edges.

Next, we detail the path we took in this thesis. We start by highlighting why an edge perspective, instead of the traditional node perspective, can add valuable insights to the Network Science field. Then we reinforce the importance of this perspective when creating or analyzing models in different fields.

1.1.1 Assessing the Importance of the Edges

When analyzing complex networks most of the attention has been given to the node centrality while less has been given to the edge centrality [11,30,42,45]. Even in the recent books on Network Science/Theory, the centrality measures presented are mostly related to the nodes.

It is known that being central, as a vertex, can have a significant impact on a network (as in spreading processes), but edges shape relations and connections and their presence/absence in specific cases can also provide valuable information about the structure and dynamics of a system.

If, on the one hand, we are interested in knowing the most powerful vertexes (with some centrality measure), on the other hand, it is intrinsic that such power only exists because edges connect nodes. And if one thinks carefully, it is through the connections that information/pathogens/cooperation propagate through the network. Because edges connect two entities, the meaning of this connection can be as broad as our imagination. Edges can represent genetic relations between species, as in the famous Darwin's Tree of Life, bridges between cities or power grids, ties of friendship or enmity, correlations between firms, and so on.

In this thesis we look at edges as a fundamental part of network analysis. We are interested in meaningful knowledge about network structure and dynamics so we start by proposing a new edge centrality measure, which we believe can give information about both. Later, we propose two models in which edge properties are crucial to defining the system behavior. More specifically, we look into signed/weighted networks where edges are labeled/weighted with information about the type of the relation between nodes.

1.1.2 Local Patterns Analysis Based on Edge Properties

It is common to look at the local properties, such as centrality measures, or global features, such as degree distributions, to characterize complex networks, but sometimes is even more important to look at something in between. In some cases it is not enough to observe nodes and edges individually. To a better understand some systems we must analyze patterns of interactions between nodes, that we can define as subgraphs or motifs.

In Social Sciences, Structural Social Balance is a global property of a signed network that is obtained by looking for the frequency of specific patterns in the network. These patterns consist of triads, cycles of length three, in which the signs of the edges correspond to positive or negative labels. These labels can shape friendship/enmity, positive/negative correlation between firms, or any other polarity relation found in complex systems.

In this thesis we show how social dilemmas and peer-influence can lead to the emergence and self-organization of social balance in signed networks. Furthermore, we also show how social balance can help to identify financial volatility. We reinforce the idea of the previous section, that the information that edges can provide is of crucial importance for behavior dynamics, by looking at these local patterns in which the most valuable information is extracted from the combination of edge characteristics.

1.1.3 Networks at Multiple Scales

There are systems that due to their complexity need to be represented as networks of networks [26, 33, 43, 48, 52, 81, 101, 104, 109, 109, 151, 188], where each node may represent another network. For

instance, networked systems can be modeled with this framework. We have a contact network where each node is a networked system and each link a source of migration/interaction between systems. Giving a specific example, we can think about bacterial populations that exchange genetic material between them. In this case, each node is a bacterial population that interacts with other populations from time to time.

Modeling and simulating this type of systems demands powerful computational frameworks. Due to the lack of real data in some of these systems, to fully understand them we need to run large-scale simulations. To develop this kind of tools, it is necessary to take into account not only the behavior and interaction models but also the goal of having efficient implementations that allow us to obtain results in a reasonable time. For this, we must use data structures and algorithms that best suit the problem in hands.

In this thesis, we propose a framework to simulate a system modeled as a network of networks. We instantiate the example of the evolution of bacterial populations over a host-contact network, where edges inside each population are shaped by mutation and recombination rates, while edges in the host-contact network are shaped by transmission probabilities. To obtain an efficient and scalable framework, we present a simulation engine based on the MapReduce programming model.

1.2 Outline and Contributions

This thesis is organized as follows. In Chapter 2, we provide the basic concepts of Network Science to a better reading of this thesis. In Chapters 3, 4 and 5, we provide proper theoretical context and methods for each contribution and describe the work already concluded. In Chapter 6, we propose new paths for the future. We also provide a list of publications and communications. Let us now detail our main contributions.

In Chapter 3 we focus on link significance in network structure, presenting a new edge-based metric for complex network analysis – spanning edge betweenness [173] – that is defined as the fraction of minimum spanning trees where a given edge is present. We consider two case studies: *i)* the confidence in phylogenetic trees [174]; and *ii)* the robustness of networks using edge percolation methods [177].

In the first case study we demonstrate that the edges chosen to build phylogenetic trees can show uncertain results since alternative edges are possible. In the second case study we observe that this metric allows to identify which edges are critical to keep a network connected, testing network connectivity with a procedure similar to edge percolation. We provide methods for the exact computation of this metric based on the well-known Kirchoff's matrix tree theorem.

In Chapter 4 we show that when analyzing complex networks, we can take advantage of studying subgraph properties instead of looking only to edges or nodes individually. Moreover, some of these

local patterns are self-organized, and are shared among different classes of networks. In this chapter, we try to identify simple self-organization principles which may be in the origin of such patterns and subgraphs. In particular, using the concept of structural balance theory, we look at small labeled motifs, more precisely cycles of size three (triads), to study dynamics in social and financial signed networks. We present three studies: *i)* a new model for the evolution of social ties, analyzing the power of peer influence [178] in the emergence of social balance; *ii)* a preliminary study about the interplay between social dilemmas and social balance; and *iii)* we analyze the relation between financial volatility and the frequency of specific motifs [168].

In the first case study, we seek the impact of peer influence in social structural balance, where the links express some positive or negative emotion between individuals. We propose a model to update social ties, based on peer influence. Our results suggest that the structural social balance observed empirically emerge for simple dynamics of peer influence [178].

In the second case study we present a preliminary study about the interplay between social dilemmas and social balance. We use both social and individual learning to solve the social dilemmas and infer the signed network. The results show that, for all dilemmas, social learning favours both the emergence of cooperation and social balance, while with individual learning the heterogeneity of the networks does not significantly affect the average strategies learned by agents neither the social balance. The results reinforce the idea that social balance emerge from peer influence mechanisms, as in cooperation.

In the third case study we analyze the relation between financial volatility and the frequency of specific motifs. Here, we present a price-correlation network model in which Standard & Poors' members are nodes connected by edges corresponding to a positive or negative price-correlations over time. We identify a close relation between volatility and the number of balanced positive triads [168].

In Chapter 5 we present a new framework for large-scale simulations of networked systems. It is difficult to analyze real complex systems when the data available is not enough to infer the behaviour of the system. Also, we can have systems that have multiple scales, networks in which each node can represent a network. The approach to study these types of real systems can be through large-scale simulations, which have always a heavy computational cost, demanding efficient solutions. Given this, we present a case study where the networked system is a host-contact network of bacterial populations. We implement an efficient and scalable simulation engine, using Wright-Fisher evolutionary model with Multi-Locus Sequence Typing (MLST) data, which allow us to simulate large populations, with millions of individuals, over real contact networks in minutes. We use the Map Reduce programming model on top of Apache Spark and GraphX API. We present results for performance scalability [175] and for population diversity, showing that fluctuations in population diversity can be explained by neutral drift only, without selection pressure, strongly depending both on the average degree and the degree heterogeneity of the host-contact network [176].

2

Network Science: Fundamental Concepts

Contents

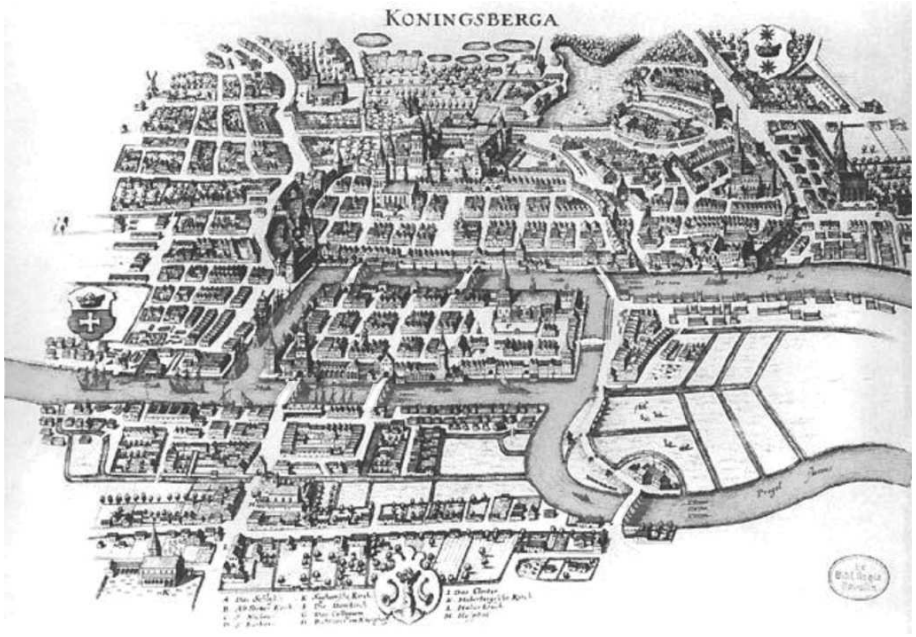
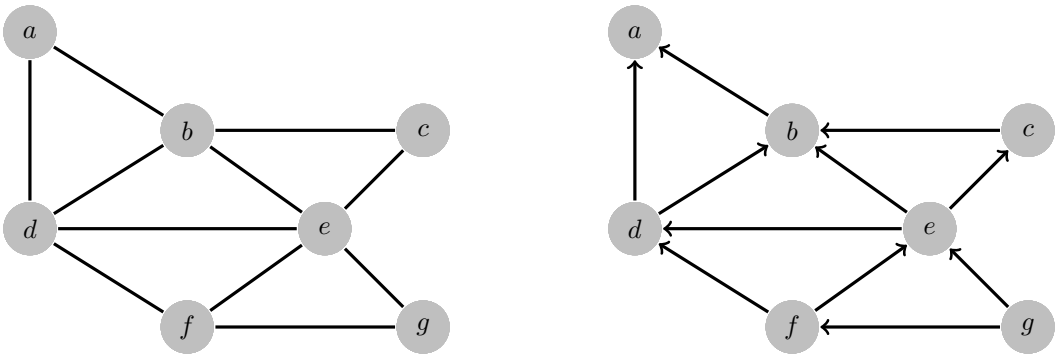


Figure 2.1: Euler's Königsberg Bridges problem: if the city is drawn as a graph with a vertex representing each part of the city and edges representing the bridges connecting the different parts, is it possible to find a path traversing each edge exactly once? (Image by von Merian-Erben – 1652, adapted from `https://www.preussenchronik.de`.)

Networks represent interactions and relations between entities. Although they portray simple and intuitive representations, a network can express very complex processes and behaviors. From species evolution, social systems or financial markets, characterizing structure and dynamics of these networks is crucial to understand different complex phenomena that occur in daily life. In this thesis, we aim to study both the structure and behaviour of real complex systems using a graph theoretical approach and knowledge about real complex networks dynamics. In the following sections, we present the basic concepts of graph theory, centrality measures and the mathematical properties of complex networks that are useful throughout this document. For more details on graph theory, complex networks theory, network science and applications we refer the reader to the books *Graph Theory*, by Diestel [53], *Complex Networks - Principles Methods and Applications*, by Vito Latora *et al.* [183], *Network Science* by Albert-László Barabási [15], and *A First Course in Network Theory*, by Ernesto Estrada and Philip A. Knight [66].

2.1 Basic Concepts of Graph Theory

Graphs are the mathematical representation of networks and graph theory is the branch of mathematics that studies the properties of a graph. We can say that graph theory started with the famous Euler's



(a) (Undirected) Graph. **(b)** Directed Graph.

Figure 2.2: Drawing Graphs: grey circles represent nodes and black lines represent edges. In 2.2a edges do not represent any order on the vertexes. For example, a is connected to d and d is connected to a. This representation means it is an undirected graph. In 2.2b edges represent order between nodes. For example, d is connected to a but the opposite is not true. Such order on the edges represents a directed graph.

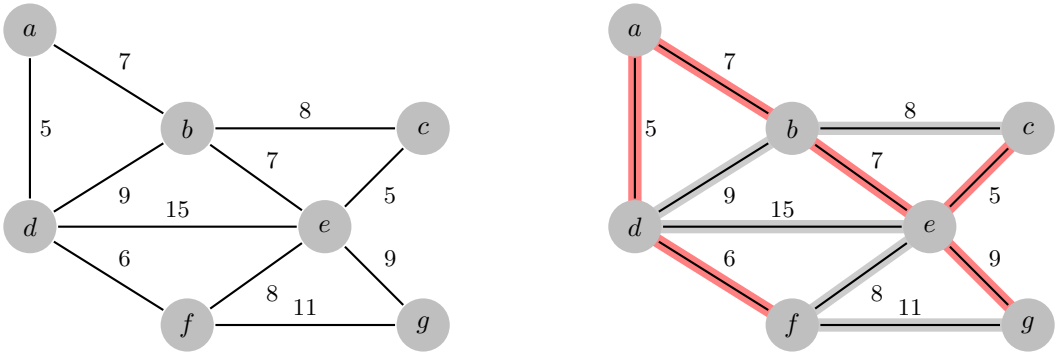
Königsberg Bridges problem (1736) [67], see Figure 2.1, but we have at least other ancient example as is the tree of life, the metaphor for phylogenetic trees used by Darwin (1861) [44] to explain the origin and relation of/between species. Next, we present the fundamental concepts of network/graph theory and some of the key ideas used throughout this thesis.

A *graph (network)*, G, is a tuple (V, E) of sets such that $E \subseteq V \times V$ where V is a set of vertexes and E is a set of edges that connect the vertexes. The size of the set E, which is also the size of the graph, is denoted by $L = |E|$ and the size of the set V is denoted by $N = |V|$. We say that a graph G is *sparse* if $|E| = O(|V|)$.

A *node/vertex* represents an entity in a graph. This entity can be a person (social networks), a pathogen/gene (biological networks), a firm (financial networks), among others. *Edges/Links* represent the connection between entities (nodes). This connections can represent physical links (a cable, a road), physical interactions (interaction between proteins), emotional ties (two people who (dis)like each other), conceptual links (dictionaries, citation networks), among others. Two vertexes u, v are *adjacent* or *neighbors* if there is an edge between them, i.e., $e = (u, v) \in E$.

A graph $G = (V, E)$ is an *undirected graph* (see Figure 2.2a) if E is a set of unordered pairs, and is considered a *directed graph* (see Figure 2.2b), or a *digraph*, if E is a set of ordered pairs. In this case an edge (u, v) is different of an edge (v, u). If a graph has multiple edges that have the same origin and destiny, is considered a *multigraph*.

A *path* in a graph G is the sequence of nodes such that each node is connected to the next node along the path by an edge. Each path consists of $n + 1$ nodes and n edges. The *distance* between two vertexes $d(u, v)$ in a graph G is the shortest number of edges in G which connects the two endpoints, u and v. The path with the shortest distance between two nodes $d(u, v)$ is called *shortest path*. If no such



(a) Weighted Graph. **(b)** A Minimum Spanning Tree.

Figure 2.3: Weighted graphs and Minimum Spanning Trees. Figure 2.3a represents a weighted graph, in which links have weights, representing some measure between the nodes. It can represent the flow of a supply chain, the cost a highway road, or even the evolutionary distance between two species. Figure 2.3b represents a minimum spanning tree built with Prim's algorithm, starting on node d.

path exists, we set $d(u, v) = \infty$. The greatest distance between any two vertexes in G is the *diameter* of G. A path with the same start and end node is a *cycle*. A cycle of length k is said to be a k*-cycle* and is denoted as C_k.

A graph G is *connected* if there is a path in G linking any two distinct vertexes, otherwise is said to be *disconnected*. Given a disconnected graph G, a *connected component* C of G is a maximal set of nodes, $C \subseteq V$, such that exists a path in C connecting any two distinct vertexes of C.

A *subgraph* of a graph $G = (V, E)$ is a graph $G' = (V', E')$ such that $V' \subseteq V$ and $E' \subseteq V' \times V'$. A *motif* is a subgraph that occurs more frequently in a graph than expected (for example, in a random graph) and represents frequent local patterns of interactions between nodes. An important concept when identifying subgraphs/motifs is graph *isomorfism*. Two graphs G and G' are *isomorphic*, denoted by $G \simeq G'$, if there is a bijection $f : V(G) \to V(G')$, such that $(u, v) \in E(G)$ if and only if $(f(u), f(v)) \in E(G')$.

A *weighted graph* is a tuple (V, E, w) where V and E form a graph $G = (V, E)$ and $w : E \to \mathbb{R}$ is a function that assigns to each edge $e \in E$ a weight $w(e)$, see Figure 2.3a. We call *signed network* to a graph G where the elements of E have a defined binary weight $w(e) \in \{-1, 1\}$ expressing a $\{$negative, positive$\}$ type of relation between the nodes.

A *tree* is a graph $G = (V, E)$ with no cycles. A *spanning tree* $T = (V, E')$ is a subgraph of G that is a tree and contains all the vertexes of G, i.e., that spans over all vertexes in V, with $|E'| = |V| - 1$. A *minimum spanning tree* (MST) is such that $\sum_{e \in E'} w(e)$ is minimum among all possible spanning trees, see Figures 2.3a and 2.3b.

A *Complete Graph* $G = (V, E)$, or a *Clique*, is a graph in which all nodes are connected to each other. Usually, if G is complete and $|V| = n$, we denote G by K_n, see Figure 2.4.

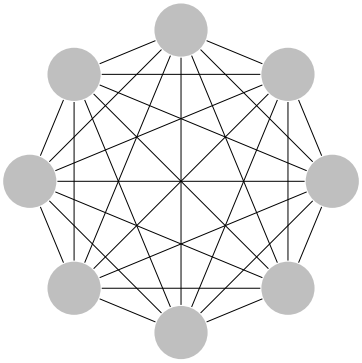


Figure 2.4: Complete Graph. Each node has an edge to all other nodes. This figure represents a K_8.

2.1.1 Representing Graphs

When manipulating graphs, depending on the operations and also on the number of nodes and edges of a graph, we may want to, or need to, represent a graph in different ways. Next, we present the most common representations. For a graphical example see Figure 2.5.

Adjacency matrix

Given a graph G, the *adjacency matrix* A is a $|V| \times |V|$ square matrix such that the entries $A_{i,j} \in \{0, 1\}$ following the rule:

$$A_{i,j} = \begin{cases} 1, & \text{if and only if } (i,j) \in E \\ 0, & \text{otherwise.} \end{cases} \tag{2.1}$$

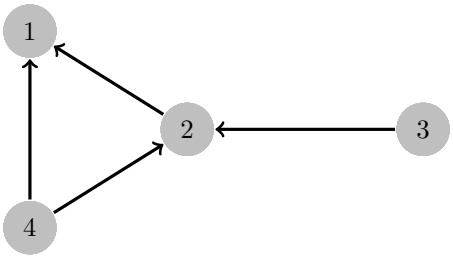
Incidence matrix

Given a graph G, the *incidence matrix* F is a $|V| \times |E|$ matrix such that the entries $F_{i,j} \in \{-1, 0, 1\}$ following the rule:

$$F_{i,j} = \begin{cases} F_{i,e} = 1 \text{ and } F_{j,e} = -1, & \text{for } e = (i,j) \in E \\ F_{i,e} = 0, & \text{otherwise.} \end{cases} \tag{2.2}$$

Edge list and Adjacency list

An *edge list* of a graph G consists in two vectors of size $|E|$, i and j, representing the position, i.e., row and column indices of the adjacency matrix, of each connected pair of vertexes. Representing a graph G with an *adjacency list* requires a vector of size $|V|$ and for each vertex a connection to list containing the neighbors.



(a) A simple graph.

$$\begin{pmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{pmatrix}$$

(b) Adjacency matrix.

$$\begin{pmatrix} 0 & 0 & -1 & 0 \\ -1 & 0 & 1 & -1 \\ 0 & -1 & 0 & 1 \\ 1 & 1 & 0 & 0 \end{pmatrix}$$

(c) Incidence matrix.

$$\begin{pmatrix} 2 & 1 \\ 4 & 1 \\ 4 & 2 \\ 3 & 2 \end{pmatrix}$$

(d) Edgelist.

Figure 2.5: Representing Graphs. Figure 2.5a is a directed graph and Figures 2.5b, 2.5c, and 2.5d are representations in the form of an Adjacency matrix, Incidence matrix and edge list.

2.2 Measures for Network Analysis

One way to characterize complex networks is to analyze local and global properties through centrality measures. The concept of centrality emerged in the context of social network analysis. Since then it has been applied in various other fields. Thus, *centrality measure* indicates the importance of a node/edge in a network. There are measures based on node degree, shortest paths and, more recently, on minimum spanning trees. Next, we present the most common centrality measures used in network analysis.

Degree centrality is a measure that represents the degree of a node, i.e., the number of edges k incident in the node. If the graph is directed, the degree of a node has two components: the number of outgoing edges (outdegree, k_{out}) and the number of ongoing edges (indegree, k_{in}). The *average degree* $\langle k \rangle$ of a graph is defined as the average node degree in the graph:

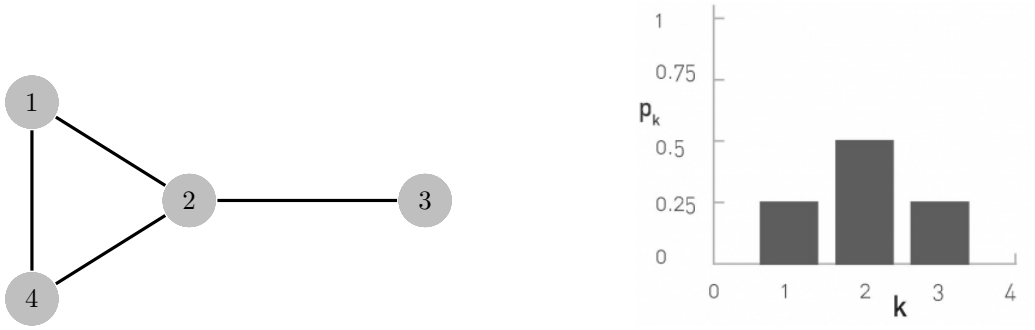
$$\langle k \rangle = N^{-1} \sum_{i=1}^{N} k_i. \tag{2.3}$$

The *degree distribution*, p_k, is the probability that a random selected node in the network has degree k:

$$p_k = \frac{N_k}{N} \tag{2.4}$$

where N_k is the number of nodes with degree k, see Figure 2.6.

Two node measures based on shortest paths are also widely used: closeness centrality and betweenness centrality. *Closeness centrality* is defined as the reciprocal of the sum of distances of the



(a) A simple graph. **(b)** Degree Distribution

Figure 2.6: Degree Centrality and Degree Distribution. Figure 2.6a represents a network in which $k_2 = 3$, $k_1 = k_4 = 2$, and $k_3 = 1$. Figure 2.6b represents the corresponding degree distribution.

shortest paths from the node to the other nodes in the graph:

$$C(i) = \frac{N-1}{\sum_{v=1}^{N} d(i,j)}. \tag{2.5}$$

A node with low distance is, on average, close to other nodes.

Betweenness centrality was created to characterize how important a node is in the communication with other nodes [80]. It is based on the assumption that information flows along shortest path so the betweenness centrality is defined has the number of shortest paths between pairs of vertexes running through a given node:

$$B(i) = \sum_{j} \sum_{k} \frac{\rho(j,i,k)}{\rho(j,k)} \tag{2.6}$$

where $\rho(j,k)$ is the number of shortest paths connecting the node j to the node k, and $\rho(j,i,k)$ is the number of these shortest paths that runs through node i.

Michelle Girvan and Mark Newman [86] extended the notion of betweenness to edges defining *edge betweenness* as the number of shortest paths between pairs of node that run along a given edge.

Clustering coefficient measures how nodes tend to cluster together, i.e, measures the total number of closed triangles in a network:

$$C = \frac{3 \times \text{number\_of\_triangles}}{\text{number\_of\_connected\_triplets}} \tag{2.7}$$

where a *connected triplet* is an ordered set of nodes ijk such that i connects to j and j connects to k.

Alternatively, the clustering coefficient may be also defined as an average over each vertex's clustering coefficient [190]. Such local clustering measure C_i of a vertex i assesses how close the neighbourhood of i is to being a clique (or a complete graph).

2.3 Random Graphs and The Scale-Free Property

Real networks are in general very large, follow a heterogeneous degree distributions [2, 6], and often portray strong clustering effects [127, 181]. Throughout this thesis we use real networks to evaluate our work, but we also use networks built on models that try to reproduce the same structure of the complex networks/systems observed in the real world. In this section we motivate and review those models.

2.3.1 Random Networks

Random network models were the first to emerge with the aim of reproducing the properties of real networks. Fixing the number of nodes and/or the number of edges, the way nodes connect are random. Two models were created, one by Erdős and Rényi [63] and other by Gilbert [84]. The two definitions are [15]:

- $G(N, L)$ Model: N labeled nodes are connected with L randomly placed edges, by Erdős and Rényi [63].
- $G(N, p)$ Model: Each pair of N labeled nodes is connected with probability p, by Gilbert [84].

Networks generated this way are truly random. As the number of nodes increase, the degree distribution follows a Poisson distribution [15]. But as real large networks became available, the characteristics identified show that real networks are not Poisson, i.e., they do not have a Poisson distribution.

2.3.2 Small-World Effect

The *small world property* become famous with the experiment by Milgram [127, 181] in the context of social networks and it is also known as *the six degrees of separation*. According to this property, two individuals, anywhere in the world, have only six or fewer acquaintances separating them, meaning that *the distance between two randomly chosen nodes in a network is short*. Given this facts, that real networks are not Poisson and that real networks show that people are closer to each other than predicted, an extension of the random network model was proposed by Duncan Watts and Steven Strogatz [190]:

- Starting with a regular ring lattice, where each node has the same number of neighbours and is connected to those that are closest, with probability p each edge is rewired to a randomly chosen node. For small p the network maintains high clustering but the random long-range edges can drastically decrease the distances between the nodes.
- For $p = 1$ all edges have been rewired, so the network turns into a random network.

With this model, Duncan Watts and Steven Strogatz achieve a better approximation in what concerns the cluster coefficient observed in real networks, but fails to explain the degree distribution. Despite the

fact that nowadays we know that random networks cannot explain most real complex systems, these studies were a deeply inspiring starting point.

2.3.3 Scale-Free Networks

In the beginning of this century, with the growth of the computational power many large real networks started to be analyzed. The World Wide Web (WWW) was sampled and mapped out by Hawoong Jeong [2]. If random networks were to represent this real network, the WWW network should have a Poisson distribution, which was not observed. Instead, the degree distribution of the World Wide Webb was well approximated by a power law distribution [2] defined as:

$$p_k = k^{-\gamma}, \tag{2.8}$$

(more details in [15]). A network with power law or broad-scale [6] degree distributions (the latter characterized by a connectivity distribution that has a power law regime followed by a sharp cutoff) have few nodes with high degree (hubs) while most nodes have a small degree. This was observed in some complex networks as protein-protein interaction, email or citation [15] networks. Because hubs can shape the system's behaviour, the so-called scale-free networks started to play an important role in the study of complex systems. In spite of perfect power law degree distributions being rare [15, 170], and may only be observed in very large (strictly speaking, infinite) networks [29, 56], models for scale-free networks have been providing a convenient framework, together with classical random graphs, to study network dynamics and topologies.

The first random scale-free network model was created by A.L. Barabási and R. Albert, the *B-A model* [13]. In the B-A model, at each time step, the network grows adding a new node and connecting it to *m* other nodes already in the network. This connections are probabilistic, depending on the degree of the nodes to be connected with, making older nodes having higher degrees, creating hubs. This is the combination of two processes – *growth* and *preferential attachment* [13].

Other model was created by Dorogovtsev-Mendes-Samukhin, called *DMS model* [56]. In this model, each time a node is added, instead of choosing other nodes to connect with, it chooses one edge randomly and connects to both ends of the edge [56]. The networks generated by the DMS model have higher cluster coefficient than those with B-A model.

In the next chapters we use regular networks and both models of scale-free networks in our experimental evaluations.

3

Link Significance in Complex Networks

Contents

The analysis of complex networks, such as social networks, biological networks, financial networks, electrical networks or even the World Wide Web, have gathered efforts from mathematicians, physicists, social and computer scientists to build several statistical measures and tools to evaluate the importance of each node and/or each link. Centrality measures are important in a large number of graph applications, from search and ranking to social and biological network analysis [41]. Most of these measures are calculated upon the nodes/vertices.

We described the most well-known in Chapter 2: degree centrality indicates the fraction of connections that a given node has over the entire network; node/edge betweenness states how important a node/edge is through the number of shortest paths between two nodes passing through it; and clustering coefficient is a key measure for social network analysis that for a given node expresses how many of its neighbours are neighbours of each other, evaluating the fraction of possible triangles in which the node is present. In this chapter, our focus is on edge significance.

The first known edge-based centrality, edge betweenness, was initially proposed by the mathematician Anthonisse and later formalized and published by Freeman in 1977 [80]. It was developed in the context of communication networks. For a given edge e it measures how central the edge is, i.e., how many shortest paths transverse that edge. In 2002, Girvan and Newman [86] applied this metric to the study of finding and evaluating community structures in networks, but little has been done in what concerns exploring new edge importance measures in a network. In 2012, Meo *et al.* [49] developed a k-path centrality, initially developed for nodes, which is based on random walks and is defined as the sum of the frequency with which a message traverses an edge e from a given source to all k-edges-distance possible destinations. These two centrality measures play a central role in reporting knowledge about data flow in a network but few about the structure/topology of the network. There are, however, other problems where alternative definitions of edge centrality are required, that should not depended on shortest paths, as is the case with the statistical evaluation of phylogenetic trees or network robustness.

When we address phylogeny, telecommunication/electric networks, among other networks, we are often interested in studying measures that go beyond shortest path/fixed distance, properties. If we want to know how strongly connected a network is, i.e., which links are fundamental to keep the network connected and which are redundant, none of the metrics described before provides direct information about that. In algorithms for phylogeny inference, we aim to validate the trees that are generated to represent evolution patterns and to identify bridges that connect different groups.

In telecommunication/electric networks we are interested in detecting which links are so crucial that if turned off could cause a breakdown, or which of them are redundant. Recently, Morone [128] presented a work in which one of the goals is to find the minimal set of nodes that, if removed, would break down the network, but once again, the work is focused on the importance of the nodes and not on the importance of the links.

The problems just described can be conveniently studied by relying on minimum spanning trees, as we will see in this chapter. First, we formally introduce a new edge centrality measure – *spanning edge betweenness* – for undirected and un/weighted graphs. This metric is defined as the fraction of Minimum Spanning Trees (MSTs) where a given edge is present. Then, we provide methods for the exact computation of this metric based on the well-known Kirchhoff's matrix tree theorem, providing experimental results in what concerns computational performance.

This new metric was initially developed due to the necessity of confidence evaluation in phylogenetic trees. Soon, other applications started to emerge, one of them being network connectivity robustness. Given this, we present two case studies. In the first case study, we evaluate how confident one can be in phylogenetic analysis with origin in algorithms like goeBURSTs [75]. The second case study is about network robustness with the goal to identify edges that are crucial both to maintain the network connected or to break the network into components rapidly.

3.1 Spanning Edge Betweenness

Let $G = (V, E)$ be a connected, undirected and weighted graph, with weight function $w : E \to \mathbb{R}$, where V is the set of vertices and $E \subset V \times V$ is the set of edges. Given an edge $e \in E$, we want to know the fraction $\delta_G(e)$ of MSTs where e occurs. The value $\delta_G(e)$ is what we call the *spanning edge betweenness* for e and it is formally defined as

$$\delta_G(e) = \frac{\tau_G(e)}{\tau_G}, \tag{3.1}$$

where τ_G is the number of different MSTs for G and $\tau_G(e)$ is the number of different MSTs for G where e occurs. Note that $\delta_G(e)$ may be zero whenever an edge e is not present in any MST, causing $\tau_G(e)$ to be zero. In what follows we write $\delta(e)$, $\tau(e)$ and τ whenever G is clear from the context.

It is clear that we can have more than one MST for a given graph G and to count how many MSTs exist in G, the solution is provided by the Kirchhoff's matrix tree theorem [95] for unweighted graphs and by Eppstein [62] for weighted graphs, where the Kirchhoff's matrix tree theorem is still used but only after some graph transformations.

3.1.1 On counting Trees

The problem of counting MSTs has been a challenge for the last decades, namely the development of efficient approaches for counting MSTs in weighted graphs, and different approaches have been described. In 1987, Gavril [82] addressed the problem of counting the number of MSTs by constructing a treelike recursive structure, the root of which is the subgraph G' formed by removing all non-maximum-

weight edges from G, and each subtree of which is constructed recursively from the components of $G \setminus G'$. The minimum spanning trees of G can then be counted by multiplying together the numbers of spanning trees at each node of this structure. This method runs in $O(nM(n))$ time, where $M(n)$ is the time required to multiply two $n \times n$ matrices. Later, in 1997, Broder and Mayr [32] improved this bound by proposing a method based on a generating function that can be expressed as a simple determinant, where the weights of the edges appear as exponents of polynomials. This method proceeds by factoring the determinant and it works for nonnegative integral edge weights. It runs in $O(M(n))$ time.

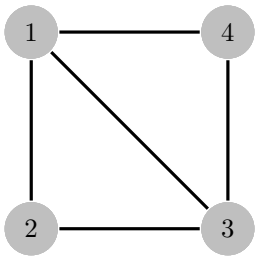
Eppstein [62] took a different approach and created the concept of equivalent graph. Specifically, one constructs from any given edge-weighted graph G an equivalent graph EG without weights, with a *sliding transformation*, such that the minimum spanning trees of G correspond one-for-one with the spanning trees of EG. Having translated the weighted graph to an equivalent unweighted graph, one can compute the number of MSTs by just applying the Kirchhoff's matrix tree theorem to the new graph.

Note that most of these approaches aim at generating and sampling MSTs, a harder problem than just counting the number of MSTs. Hence, although we use some of these ideas in our approach, since we are just counting MSTs, we have a less complex approach and are thus able to achieve better performance. Moreover, our approach may be applied to the general case of graphical matroids. Note that the problem of finding an MST is a particular case of graphic matroids [139] and, thus, finding a solution for a given graph consists of solving an instance of graphic matroids [139, 182, 191], which can be optimally solved with a greedy approach [61]. One of those greedy approaches is precisely the Kruskal's algorithm [112]. In the general case of graphic matroids, edges may be unweighted, which is usually the case. We just need to define a total order for the edges based on specific criteria which is precisely what we have in general phylogenetic studies based on trees [75]. Contrary to other methods that depend on edges being weighted, our approach just depends on sorting edges in increasing order and, thus, we just require a total order to be defined.

3.2 Implementation

We want to compute, as efficiently as possible, the spanning edge betweenness $\tau_G(e)$ for a given $e \in E$, where $G = (V, E)$ is a connected, undirected and weighted graph.

We will start by showing how to compute $\tau_G(e)$ and $\delta_G(e)$ when $G = (V, E)$ is a connected, undirected and unweighted graph, with $n = |V|$ vertices and $m = |E|$ edges. Note that in this case the number τ of MSTs in G is equal to the number of spanning trees in G and it can be computed directly from the Kirchhoff's matrix tree theorem [107]. Then we will extend our result to weighted graphs and present some experimental results.



(a)

$$L = \begin{pmatrix} 3 & -1 & -1 & -1 \\ -1 & 2 & -1 & 0 \\ -1 & -1 & 3 & -1 \\ -1 & 0 & -1 & 2 \end{pmatrix}$$

(b)

$$L^* = \begin{pmatrix} 2 & -1 & 0 \\ -1 & 3 & -1 \\ 0 & -1 & 2 \end{pmatrix}$$

(c)

$$det(L^*) = |L^*| = \begin{vmatrix} 2 & -1 & 0 \\ -1 & 3 & -1 \\ 0 & -1 & 2 \end{vmatrix} = 8$$

(d)

Figure 3.1: Kirchhoff's Theorem for unweighted and undirected graphs.

3.2.1 Unweighted graphs

Let $F \in \{-1, 0, 1\}^{n \times m}$ be the incidence matrix for G such that $F_{i,e} = 1$ and $F_{j,e} = -1$, for $e = (i, j) \in E$. Let us also consider the reduced incidence matrix $F^{(i)}$ obtained from F by deleting row i. Note that $\mathrm{rank}(F) = n - 1$, $\mathrm{rank}(F^{(i)}) = n - 1$, and the determinant for any square submatrix of $F^{(i)}$, for any i, is either 0, -1, or 1. A more interesting observation due to Kirchhoff is that a submatrix $(n-1) \times (n-1)$ of $F^{(i)}$, for any i, is non-singular if and only if its columns correspond to the edges of a spanning tree.

Theorem 1 (Kirchhoff [107])**.** *The spanning trees of a connected and undirected graph G with n vertices are the non-singular $(n-1) \times (n-1)$ submatrices of the reduced incidence matrix $F^{(i)}$, for any i, and the determinants of the submatrices are all ± 1.*

Hence, by using Cauchy-Binet theorem on determinants, the number of spanning trees τ is given by the Kirchhoff's well known formula

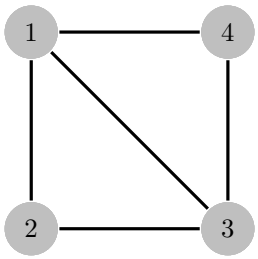
$$\tau = \det(L^{(i)}) \tag{3.2}$$

$$= \sum_S \det(F_S^{(i)}) \det({F_S^{(i)}}^\top) \tag{3.3}$$

$$= \sum_S \det(F_S^{(i)})^2, \tag{3.4}$$

where S ranges over the subsets of E with size $n-1$, $L = FF^\top$ is the Laplacian matrix for G, and $L^{(i)}$ denotes the matrix obtained from L by deleting row and column i, see Figure 3.1.

We extend this result to compute $\tau(e)$, for $e \in E$, as follows.



(a)

$$L = \begin{pmatrix} 3 & -1 & -1 & -1 \\ -1 & 2 & -1 & 0 \\ -1 & -1 & 3 & -1 \\ -1 & 0 & -1 & 2 \end{pmatrix}$$

(b)

$$L^{(1,2)} = \begin{pmatrix} 3 & -1 \\ -1 & 2 \end{pmatrix}$$

(c)

$$det(L^{1,2}) = |L^{(1,2)}| = \begin{vmatrix} 3 & -1 \\ -1 & 2 \end{vmatrix} = 5$$

(d)

Figure 3.2: Spanning Edge Betweenness. Given the example of Figure 3.1, we now calculate the number of spanning trees in which a given edge e is present. We illustrate the example for $e = (1, 2)$, deleting the corresponding indexes of rows and columns. Its spanning edge betweenness value is now possible to calculate: $\delta(1, 2) = 5/8$.

Theorem 2. *Given $G = (V, E)$ an undirected and connected graph, let $e = (i, j) \in E$ and $L^{(ij)}$ denote the matrix obtained from L by deleting rows i and j and columns i and j. Then, $\det\left(L^{(ij)}\right)$ is the number of spanning trees $\tau(e)$ that contain e.*

Proof. As discussed above, the total number of spanning trees is given by $\det(L^{(i)})$, for any i. Let G' be the graph where we remove the edge (i, j) and L' be the Laplacian for G'. Hence, the total number of spanning trees for G' is given by $\det(L'^{(i)})$, for any i, and the number of MSTs that contain (i, j) is simply given by $\det(L^{(i)}) - \det(L'^{(i)})$. Let us show that $\det(L^{(ij)}) = \det(L^{(i)}) - \det(L'^{(i)})$ or, equivalently, that $\det(L^{(i)}) = \det(L'^{(i)}) + \det(L^{(ij)})$. We have that $L^{(i)} = F^{(i)}{F^{(i)}}^\top$ and $L^{(ij)} = F^{(i,j)}{F^{(i,j)}}^\top$, where $F^{(i,j)}$ is obtained from F by removing rows i and j, and, using Cauchy-Binet's formula, we can show instead that

$$\sum_S \det\left(F_S^{(i)}\right)^2 = \sum_{S'} \det\left(F'^{(i)}_{S'}\right)^2 + \sum_{S^*} \det\left(F^{(i,j)}_{S^*}\right)^2 \tag{3.5}$$

where F' is the incidence matrix for G', S ranges over the subsets of E with size $n-1$, S' ranges over the subsets of $E \setminus \{(i, j)\}$ with size $n-1$, and S^* ranges over the subsets of E with size $n-2$. Since S' ranges over the subsets of $E \setminus \{(i, j)\}$, we can replace F' by F in previous equation. Note also that

$$\sum_{S^*} \det\left(F^{(i,j)}_{S^*}\right)^2 = \sum_{S^* \cup \{(i,j)\}} \left(\det\left(F^{(i,j)}_{S^*}\right) \times \pm 1\right)^2 \tag{3.6}$$

$$= \sum_{S^* \cup \{(i,j)\}} \det\left(F^{(i)}_{(S^* \cup \{(i,j)\})}\right)^2 \tag{3.7}$$

because adding edge (i,j) to S^* and considering $F^{(i)}$ instead of $F^{(i,j)}$ just adds a term ± 1 to each matrix determinant. Therefore,

$$\sum_{S} \det\left(F_S^{(i)}\right)^2 = \sum_{S'} \det\left(F_{S'}^{(i)}\right)^2 + \sum_{S^* \cup \{(i,j)\}} \det\left(F_{(S^* \cup \{(i,j)\})}^{(i)}\right)^2 \tag{3.8}$$

which is an equality as the first term on the right side ranges over all subsets of E with size $n-1$ that do not contain (i,j) and the second term ranges over all subsets of E with size $n-1$ that do contain (i,j). □

Hence, using both results, we can easily compute $\delta(e)$ for any $e \in E$. Note also that the same is true for multigraphs, graphs that allow multiple edges between the same pair of vertices, as both results above hold with the following changes in the Laplacian matrix L [118]: if vertex i is adjacent to vertex j in G, then L_{ij} is equal to the number of edges between i and j; when counting the degree of a vertex, all loops are excluded.

3.2.2 Weighted graphs

Let $G = (V, E)$ be a connected, undirected and weighted graph, with weight function $w : E \to \mathbb{R}$. We can compute a MST for G by using the Kruskal's algorithm [112]:

1. sort E with respect to w in increasing order;
2. create a forest M where each $u \in V$ is a tree;
3. iterate over E in increasing order and, for each $(u,v) \in E$, if u and v are in different trees, add (u,v) to M combining both trees as single tree;
4. return M.

Note that we may get different MSTs by changing the order obtained in step 1, where we can exchange positions of edges with the same weight. In particular, since it is well known that the sorted list of edge weights is the same for any MST, changing the order allow us to obtain all different MSTs.

We can take this a step further. Let $e \in E$ and let M' be the forest obtained in Kruskal's algorithm after processing all edges $e' \in E$ such that $w(e') < w(e)$. Let also G' be a graph where each tree in M' is a vertex and where we add all edges in E with weight $w(e)$. Note that G' may be a multigraph and, since all edges have the same weight, we may look at it as an unweighted multigraph. Moreover, if we consider the connected component C of G' that contains edge e, and by using results in previous section, we can compute the number τ_C of spanning trees for that component and also the number $\tau_C(e)$

of spanning trees for that component where e occurs. The key observations are that we can use this approach to compute the number of spanning trees in G and that $\delta_G(e) = \delta_C(e)$.

It is clear that an edge $e \in E$ can only permute with another edge $e' \in E$ to form a different MST iff $w(e) = w(e')$ and, if a MST M contains e, adding e' to M leads to a cycle. Moreover, that cycle can only contain edges with weight equal or lower than $w(e)$, otherwise M would not be an MST. If we add all edges with weight $w(e)$ to M and contract all edges with weight lower than $w(e)$, we obtain the graph G' and the product of the number of trees in each connected component of G' is the number of ways we can select edges with weight $w(e)$ for each MST of G. By doing this for each different weight in G and then multiplying all values, we obtain the number of MSTs τ for G.

Since a given edge e only has influence on the number of trees for the component of G' where it occurs and the number of trees for all other components and weights remain the same, it follows that $\delta_G(e) = \delta_C(e)$.

Hence, given a connected, undirected and weighted graph $G = (V, E)$, with weight function $w : E \to \mathbb{R}$, we can compute the number of MSTs and the spanning edge betweenness for each edge as follows:

1. sort E with respect to w in increasing order;
2. let $H = (V, \emptyset)$ and $\tau_G = 1$;
3. iterate over E in increasing order and, while edges have the same weights, add them to H;
4. for each connected component C in H, compute τ_C using Theorem 1, update $\tau_G = \tau_G \times \tau_C$, and, for each edge $e \in C$, compute $\tau_C(e)$ using Theorem 2 and $\delta_C(e)$ using Equation 3.1;
5. contract all edges in H such that each connected component becomes a single vertex;
6. if H has more than one vertex, repeat from step 3, otherwise return τ_G.

3.3 Towards Link Confidence in Phylogenetic Analysis

3.3.1 Motivation

The use of trees for phylogenetic representations started in the middle of the 19th century. One of their most popular uses is Charles Darwin's sole illustration in "The Origin of Species" [44] (see Figure 3.3). The simplicity of the tree representation makes it still the method of choice today to easily convey the diversification and relationships between species. But, for the research community, it may be unclear that the presented tree is just an hypothesis, chosen among many possible alternatives. In this scenario, it is important to quantify our confidence in both the trees and the branches or edges included in such trees.

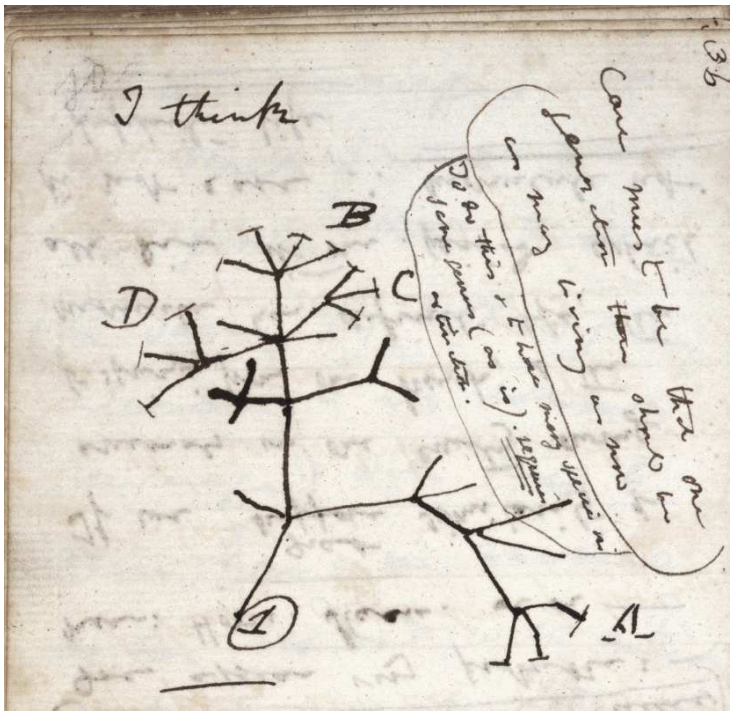


Figure 3.3: Tree of Life. Charles Darwin's 1837 first diagram of an evolutionary tree sketch, from his First Notebook on Transmutation of Species (1837), adapted from `https://en.wikipedia.org/wiki/Tree_of_life_(biology)`.

Although MST computation is a classical mathematical problem and its application to evolutionary studies had already been suggested more than a decade ago [69], it was not until recently, with the advent of multilocus sequence typing (MLST) [125] and particularly whole genome sequencing, that they gained popularity has an alternative to eBURST [73]. One appeal of MSTs is the simplicity of their assumptions that reflect the concept of minimal evolution. MSTs simply link together the more closely related individuals in the population, generating a single tree representing all individuals. The Steiner trees [166], generated by the more classical methods for phylogenetic inference, place individuals exclusively in branch tips. By allowing individuals to be placed in interior nodes, spanning trees and MSTs in particular, may better convey the peculiarities of short-term intraspecific evolution [69].

It was also recently pointed out that the optimal implementation of the BURST rules in goeBURST, results in a set of disjoint MSTs [75]. Analyses of trees generated using multilocus sequence typing data (MLST) and the goeBURST algorithm revealed that the space of possible MSTs in real data sets is extremely large. Selection of the edge to be represented using bootstrap could lead to unreliable results since alternative edges are present in the same fraction of equivalent MSTs. The choice of the MST to be presented, results from criteria implemented in the algorithm that must be based in biologically plausible models. The fact that a single tree is reported from a multitude of possible and equally optimal solutions and that no statistical metrics exist to evaluate them, justified a new heuristic approach – spanning edge

betweenness – to address these issues [154]. In this context, we can infer the following information:

1. if spanning edge betweenness is 1 we have 100% of confidence in the relation between the two locus represented by the phylogenetic tree;
2. if it is < 1 this means that the we cannot be absolutely certain about that direct relation.

Spanning Edge Betweenness in PHYLOViZ

We have implemented our metric as a module for PHYLOViZ [76], available at `http://www.phyloviz.net/`, for the evaluation of phylogenetic confidence. Our implementation uses the Colt library[1] for linear algebra operations, including in particular the computation of matrix determinants. Since we are dealing with relatively large sparse graphs, we use the class `SparseDoubleMatrix2D` in Colt. We also use a disjoint-set data structure to track connected components similarly to what is common in Kruskal's algorithm implementations [40].

The time complexity of the proposed approach is dominated by the time required to compute the determinants, since the Kruskal's runs in $O(m \log n)$ time, for a graph with n vertices and m edges. Computing the determinant for a $n \times n$ matrix can be done in $O(n^{3/2})$ time [83]. Hence, for sparse graphs with $m = O(n)$, this method runs in $O(n^{2.5})$ time since we have to compute a determinant for each edge. In practice, it runs faster as connected components are usually much smaller than the original graph.

Computational Performance

As the first approach to spanning edge betweenness was in the context of phylogenetic trees, our performance evaluation was performed in the same context with MLST data. In this context, is convenient to remember that each ST correspond to a vertex and each SLV connection to an edge.

The performance analysis was conducted using an Intel i7 a 2.3GHz, with 6GB of RAM in nine available MLST databases of important human pathogens: *Burkholderia pseudomallei*, *Campylobacter jejuni*, *Enterococcus faecium*, *Haemophilus influenzae*, *Neisseria spp.*, *Pseudomonas aeruginosa*, *Streptococcus agalactiae*, *Staphylococcus aureus*, and *Streptococcus pneumoniae*. These databases were retrieved on June 24th, 2014, from public repositories available in different websites. In Table 3.1 we provide details for the nine different datasets and the running times to compute spanning edge betweenness for all edges in phylogenetic trees computed by PHYLOViZ. Our solution allows the algorithm to run on a common laptop in reasonable time. For data sets with few STs, such as *H. Influenzar* and *P. aeruginosa*, it takes less than 2 seconds. However, for datasets with a larger number of STs, such as *C. jenuni*, it can take almost one hour. The running time will depend, mostly, on the number of STs of each

[1] `http://acs.lbl.gov/software/colt/`

Data sets	Statistics for largest CC					
Species	**# STs**	**# Edges**	**# MSTs**	**Compactness**	**Clustering**	**Running Time (s)**
B. pseudomallei	624	1476	$10^{276,74}$	0,008	0,283	16,8
C. jejuni	2318	9288	$10^{1440,45}$	0,003	0,600	2759,3
E. faecium	610	1906	$10^{338,32}$	0,010	0,464	23,2
H. influenzae	150	668	$10^{94,31}$	0,059	0,678	1,6
Neisseria spp.	2011	12701	$10^{1521,63}$	0,006	0,627	1489,9
P. aeruginosa	101	159	$10^{22,81}$	0,031	0,442	2,0
S. agalactiae	519	2520	$10^{365,79}$	0,019	0,690	13,1
S. aureus	1089	8317	$10^{970,83}$	0,014	0,796	277,2
S. pneumoniae	1275	5203	$10^{788,28}$	0,006	0,641	362,5

Table 3.1: Statistics relative to the largest CC linking STs at the SLV level. Columns represent the number of STs, the number of edges, the total number of possible MSTs, the compactness and clustering indexes, and the algorithm running time in seconds.

data set, that is clearly related to the dimensions of the matrix representing the relationships between STs. Hence, the number of STs is directly related to the number of operations required to calculate determinants.

A more efficient implementation

Beside the motivation of an implementation of a module to PHYLOViZ application, we also implemented an offline version of the module where we used some extra settings to accelerate its execution and allow to obtain results that are not meant to be shown in PHYLOViZ. In this offline implementation, we used MTJ (Matrix Toolkit Java)[2] library that is is a high-performance library for developing linear algebra applications. MTJ is based on BLAS[3] and LAPACK[4] for its dense and structured sparse computations.

With this library, we use the LU decomposition to calculate the determinant. We create an upper triangle dense matrix and then we go through all the elements of the diagonal. Instead of multiplying all the determinants as in the module developed for PHYLOViZ we sum the logarithm of each absolute diagonal value, obtaining instead the logarithm of the determinant.

To improve our running time, we used the Java concurrent library for computing edges statistics in parallel. Since the statistics for each edge can be computed independently, we could parallelize statistics computation in a straightforward manner. Package available at `https://bitbucket.org/phyloviz/popsim-analysis`

3.3.2 Experimental Analysis

Why this particular tree? This is a recurring question for most researchers on phylogenetic studies. Although methods and tools use well known rules for the construction of phylogenetic trees, there can

[2] `https://github.com/fommil/matrix-toolkits-java/`
[3] `http://www.netlib.org/blas/`
[4] `http://www.netlib.org/lapack/`

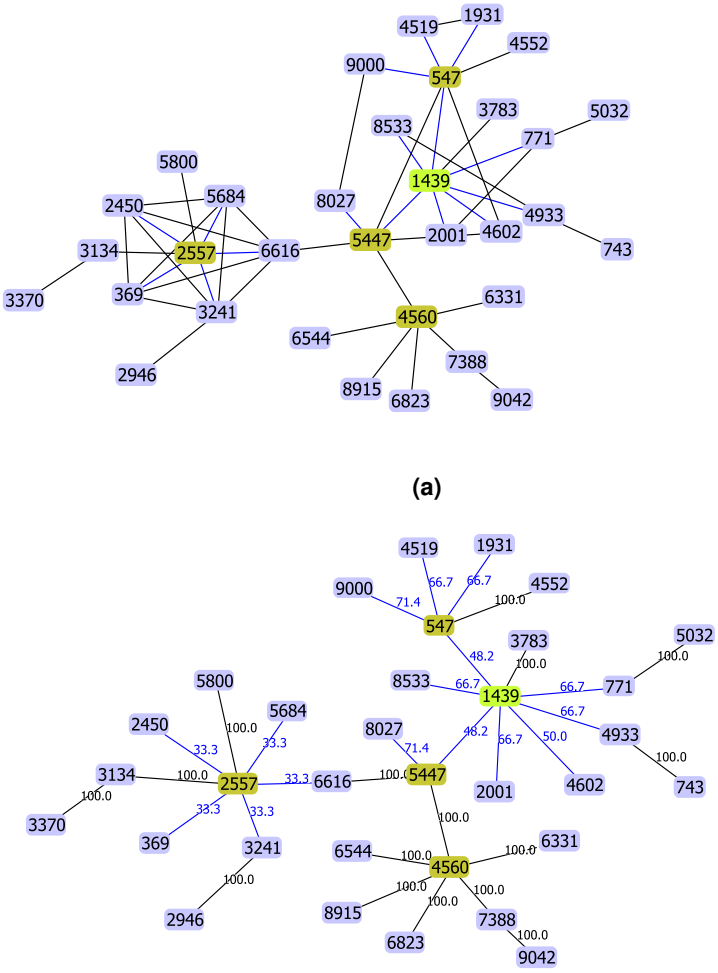


Figure 3.4: Representation of *S. pneumoniae* CC 28 using PHYLOViZ. 3.4a Representation of all edges linking STs at SLV level. 3.4b Representation of edges from the MST selected after application of goeBURST rules, with the spanning edge betweenness for each edge.

be multiple trees satisfying these rules. Here, we provide a simple metric that allows us to know how many trees satisfy our inference rules and, once we compute one such trees, how confident can we be on each edge included. In Figure 3.4, we can draw an important conclusion from calculating spanning edge betweenness which is that all alternative edges linking a given ST to other STs are present in the same proportion of equivalent MSTs. For instance, ST1931 is linked by two equivalent edges to ST4519 and to ST547 (Figure 3.4a), and both edges are represented in 66.7% of all equivalent MSTs (see the chosen edge by goeBURST in Figure 3.4b). This means that the mark-recapture/bootstrap approach suggested previously to select between alternative equivalent edges [154], cannot provide an adequate solution to this problem. In the space of MSTs, equivalent edges are equally represented and so any dominance of a given edge in the bootstrap procedure will be a consequence of the limitations of the

Data sets	Statistics for largest CC								
Species	**SLV**			**DLV**			**TLV**		
	#STs	**# Edges**	**#MSTs**	**#STs**	**# Edges**	**#MSTs**	**#STs**	**# Edges**	**#MSTs**
B. pseudomallei	624	1476	$10^{264,87}$	944	12017	$10^{460,78}$	1020	56776	$10^{517,07}$
C. jejuni	2318	9288	$10^{1361,82}$	3532	101377	$10^{2163,62}$	6397	539253	$10^{3577,54}$
P. aeruginosa	101	159	$10^{9,50}$	799	3078	$10^{215,07}$	1382	16122	$10^{466,81}$
S. agalactiae	519	2520	$10^{336,78}$	651	17955	$10^{456,87}$	651	36517	$10^{456,87}$

Table 3.2: Statistics relative to the largest CC linking STs at the SLV, DLV and TLV levels of construction. SLV means that the graph contains only links at a distance of one, DLV until a distance of two and TLV until a distance of three.

procedure and should not be used as selection criterion. The choice between alternative edges must be based on well defined criteria that should reflect an underlying model of microbial evolution. The goeBURST rules have such an underlying model [73, 75, 165] and offer a robust method to select a tree from equivalent MSTs.

Going further in our analysis and using the datasets described in Table 3.1 we determined the goeBURST forest of each species by linking STs at the single-locus variant (SLV) level, double-locus variant (DLV) level and triple-locus variant (TLV) level. Unless otherwise stated, the analyses were performed on the forest generated by creating trees linking STs at the SLV level.

We calculated the number of possible MSTs in the largest CC of each of these species. The information is present in Table 3.2. As expected, even only for the largest CC, the number of possible MSTs is quite large, in fact it exceeds a googol [103] for most of the species considered. When MST results are presented, a single tree is shown, chosen from among this universe of possible trees following a set of rules or simply as a consequence of the algorithm used and the input order of the nodes [154]. The goeBURST algorithm implemented in PHYLOViZ, selects the final tree according to a set of well defined rules that guarantee the uniqueness and consistency of the selected tree [75, 76]. The impact of the application of each of the rules on the universe of possible trees for the largest CC of each species is presented in Table 3.2. For most species, a single tree is obtained when applying up to the second tiebreak rule (higher number of DLVs), but in the case of *B. pseudomallei*, *C. jejuni* and *Neisseria spp.* a single tree is only obtained when invoking rules up to the third tiebreak rule (higher number of TLVs). In the case of *S. pneumoniae* only the last tiebreak rule (higher number of STID) results in a single tree. Large reductions in the available tree space can be seen with the application of each goeBURST rule and this can be used to evaluate the impact of each rule on the final phylogenetic hypothesis proposed by the algorithm.

The actual reduction of tree space varies between the species considered and can be seen in Table 3.3. It is clear that the number of STs influences the number of possible trees, but this relationship is complex, with the number of possible edges linking STs at the SLV level, having a similar and equally significant influence on tree space (Table 3.1). For instance, when comparing the largest CCs of *B. pseu-*

Data sets	Rules for biggest CC					
Species	**No Rules**	**SLV**	**DLV**	**TLV**	**Frequency**	**STID**
B. pseudomallei	$10^{264,87}$	$10^{196,10}$	$10^{6,06}$	1	1	1
C. jejuni	$10^{1361,82}$	$10^{635,50}$	$10^{5,08}$	1	1	1
E. faecium	$10^{327,03}$	$10^{206,29}$	1	1	1	1
H. influenzae	$10^{81,48}$	$10^{3,17}$	1	1	1	1
Neisseria spp.	$10^{1411,38}$	$10^{286,33}$	$10^{1,20}$	1	1	1
P. aeruginosa	$10^{9,50}$	$10^{6,77}$	1	1	1	1
S. agalactiae	$10^{336,78}$	$10^{42,50}$	1	1	1	1
S. aureus	$10^{907,23}$	$10^{49,58}$	1	1	1	1
S. pneumoniae	$10^{755,48}$	$10^{181,88}$	$10^{2,16}$	$10^{0,60}$	$10^{0,60}$	1

Table 3.3: goeBURST breaking rules effect. Each column represents the number of possible MSTs after each break rule.

Data sets	Biggest CC								
Species	**SLV**			**DLV**			**TLV**		
	#STs	**# Edges**	**#MSTs**	**#STs**	**# Edges**	**#MSTs**	**#STs**	**# Edges**	**#MSTs**
B. pseudomallei	596	1418	$10^{264,87}$	944	12017	$10^{460,78}$	1020	56776	$10^{517,07}$
C. jejuni	2224	8700	$10^{1361,82}$	3532	101377	$10^{2163,62}$	6397	539253	$10^{3577,54}$
P. aeruginosa	65	87	$10^{9,50}$	799	3078	$10^{215,07}$	1382	16122	$10^{466,81}$
S. agalactiae	492	2279	$10^{336,78}$	651	17955	$10^{456,87}$	651	36517	$10^{456,87}$

Table 3.4: Statistics of the Graphs at each Level of construction SLV means that the graph contains only links at a distance of one, DLV until a distance of two and TLV until a distance of three.

domallei and *E. faecium*, although both have a similar number of STs, the latter has a higher number of possible edges and trees. An even more striking example is the comparison between the largest CCs of *S. aureus* and *S. pneumoniae*, with the former having a smaller number of STs, but a higher number of possible edges and trees (Table 3.1).The measurements of compactness and clustering of the tree of the largest CC capture properties that may be related to intrinsic characteristics of each species. For instance, values of compactness less then 0.010 are associated with the species listed above; those that also reach higher tiebreak rules to identify a single tree (Table 3.1). These species are known to have very high rates of recombination [110, 129, 141, 192] and the existence of recombination can generate STs with multiple possible pathways of descent, which in turn would be expected to affect a graph's compactness.

The goeBURST algorithm in PHYLOViZ can be run by creating sets of disjoint trees linking STs at DLV or TLV level and the result of this analysis for the largest CC of the species considered is presented in Table 3.4. As expected, a higher number of STs and possible edges in the largest CC, as we go from SLV to TLV, results in higher numbers of possible trees. The tree space at any given level, when considering the entire forest, is the product of the number of trees for each CC and is greatly influenced by the largest CC, hence our decision to present the analysis of the largest CC only for simplicity.

We have previously proposed that the tiebreak rule reached before deciding if an edge should be drawn, could be used to evaluate the reliability of the represented hypothetical pattern of descent [75].

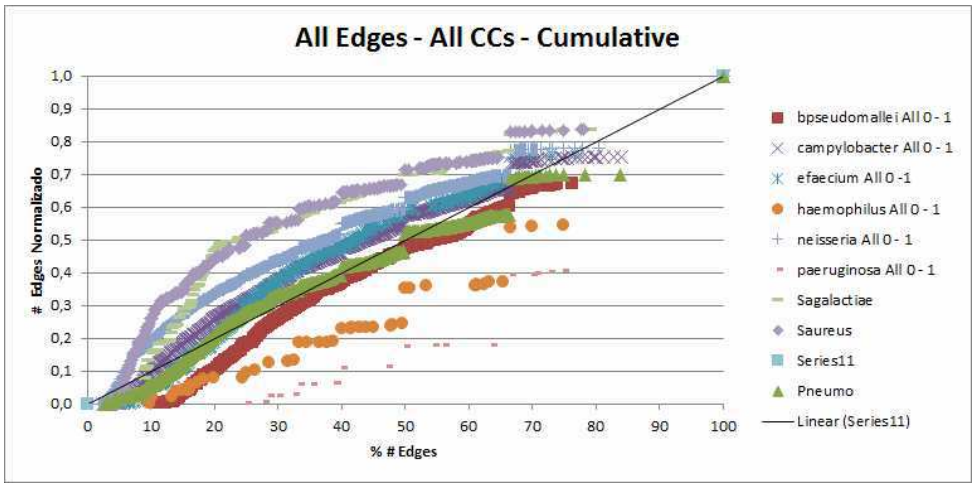


Figure 3.5: Cumulative distribution of the spanning edge betweenness of all Edges in all CCs.

The spanning edge betweenness can be used for the same purpose, with results that are similar to the bootstrap procedure more commonly used [74].

In Figure 3.5 we show the cumulative distribution of the spanning edge betweenness of all edges, for all the nine species, in the set of all CCs, calculated at the SLV level by the goeBURST algorithm in PHYLOViZ. The distribution of spanning edge betweenness of the edges of the MST selected by goeBURST is variable between species. In contrast to the number of MSTs discussed previously, there is not a dominant role of recombination in determining the shape of the distribution, since the species identified previously as being recombinogenic are not homogeneous in their distributions. These differences possibly reflect a more complex interplay of the intrinsic properties of each species, such as mutation and recombination rates and possibly their ratio.

3.4 On Network Connectivity Robustness

3.4.1 Motivation

Minimum spanning trees have been used for decades for network design, cluster analysis, among others. When constructing certain networks – such as electrical, computer, transportation, and telecommunication networks – the major concern is to choose the cheaper path for laying the connections. On the other hand, if we already have a network, how can we identify which are the links whose presence is imperative to connect all the nodes and which provide a more flexible choice? On other perspective: given a computer network, which connections should we choose to assure its connectivity preventing a massive disruption? Which connections/edges are critical?

Spanning edge betweenness allows us to give some answers for these questions. When analyzing

minimum spanning trees, shortest-paths or random walks approaches yield insufficient information to infer the connectivity robustness of a network or how redundant are some connections, depending on the subject in study. The fact that spanning edge betweenness gives directly the probability of an edge being in a minimum spanning tree, thus reflecting how important it is for the network structure, it ensures a high confidence in the analysis of network connectivity and edge redundancy. In this context We can infer the following information:

1. if spanning edge betweenness is 1 than the edge has to be on the network to keep it connected;
2. if it is 0, which only can occur in weighted networks, if we consider only optimal trees, than the edge is completely redundant;
3. being the value between 0 and 1 it means that there are alternative paths that maintain the network connected, thus expressing the redundancy of an edge.

We are interested in analyzing two main aspects related to spanning edge betweenness: *(i)* if the distribution of the values of spanning edge betweenness gives specific information about the topology of a given network; *(ii)* if with an edge percolation method spanning edge betweenness leads to a faster disruption in the networks, rapidly increasing the number of components of the networks, when compared with the most used edge measure, edge betweenness.

In both cases we start by calculating both spanning edge betweenness and edge betweenness for each edge. A generic implementation of spanning edge betweenness can be found at `https://bitbucket.org/phyloviz/popsim-analysis`.

3.4.2 Experimental Analysis

To evaluate the significance of the spanning edge betweenness in network connectivity we chose eight different networks, with different sizes and from different contexts. Four are real well-known networks (Karate, Power Grid, Political Blogs and NetScience)[5], and four are random networks: two generated from B-A model [13] and two networks with community structure[6]. The properties of these networks are in Tables 3.5, 3.6 and 3.7.

Network	# Nodes	# Edges
Karate	34	78
PowerGrid	4941	6594
Polblogs	1490	2742
NetScience	1589	1252

Table 3.5: Statistical details for real networks.

[5] `http://www-personal.umich.edu/~mejn/netdata/`
[6] `https://sites.google.com/site/santofortunato/inthepress2`

# Nodes	# Edges	Average Degree
1000	2975	4
1000	4939	4

Table 3.6: Barabási-Albert model parameters for generating random networks.

# Nodes	# Edges	Min / Max Degree	Min / Max Community Size
1000	2222	4 / 8	20 / 40
1000	3985	8 / 16	20 / 40

Table 3.7: Model parameters for generating random networks with community structure.

Our analysis start with the computation of five measures: node degree centrality, node betweenness, edge betweenness, cluster coefficient and spanning edge betweenness. Our goal is to see if there is any correlation between spanning edge betweenness and the other well-known network measures. For this we calculated the correlation between Spanning edge betweenness and edge betweenness directly, both are edge measures edges. To correlate spanning edge betweenness with the other node-based metrics – node betweenness, degree centrality and cluster coefficient – we calculated the correlation between them taking into account the minimum/maximum/average metrics between the source and destination nodes of each edge.

The first conclusion is that spanning edge betweenness has no correlation with the other measures. None the correlations showed meaningful values. This reinforces the idea that this measure provides novel information that was not given before. In Figure 3.6, we show that spanning edge betweenness has a different expression than edge betweenness.

While spanning edge betweenness took values between 0 and 1, expressing directly the importance of an edge, edge betweenness took all of its values below 0.3. Comparing directly both measures, it is possible to see that the values of edge betweenness do not allow to infer clear information about network structure. Edge betweenness is about how much information flow passes through an edge in shortest paths, while spanning edge betweenness is about the significance of an edge in network connectivity, potentially identifying edges that can break the network and reflecting if the network has a strong or weak redundancy. We can also see that PowerGrid has a very different behaviour from other three chosen networks. This is because the topology of the network is like a tree, or a star, with only a few redundant edges, being one example that if a link is disconnected, most probably the network will break. The other networks illustrate the redundancy that is expected from that kind of networks. As friends are friends with each other, as one cites another, there can be much alternatives to maintain the network connected and reachable between all nodes.

To reinforce the idea that spanning edge betweenness provides information about the connectivity robustness of a network, we also present an evaluation with edge percolation. We show how removing edges from a network affects its structure. Has a baseline we start by removing edges randomly.

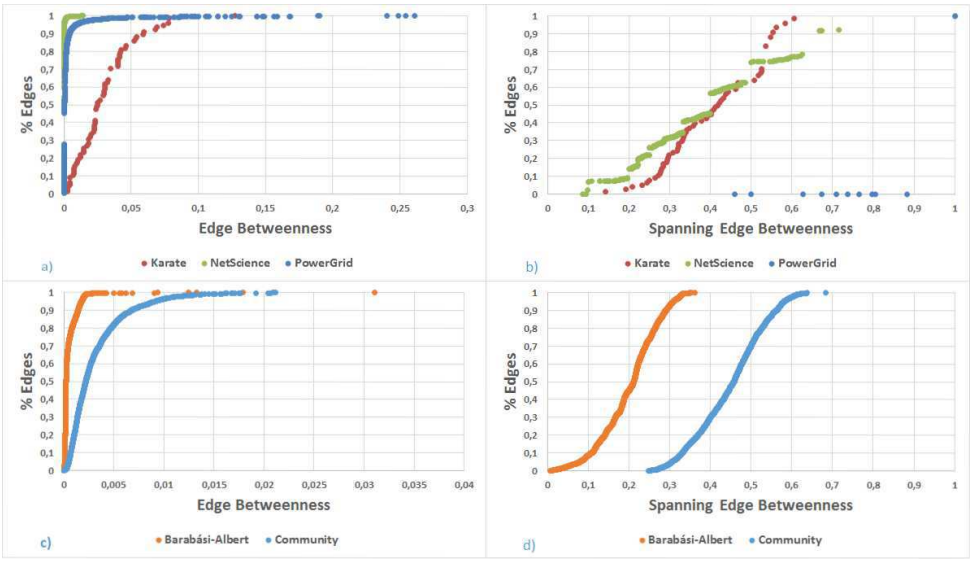


Figure 3.6: Edge Betweenness Vs Spanning Edge Betweenness. In panels a) and c) we show the values of edge betweenness for three empirical networks and two random generated networks. In panels b) and d) we show the values of spanning edge betweenness for the same networks. While spanning edge betweenness shows a wide range of values, expressing edge significance in network structure, edge betweenness is limited to a very small set of values not being possible to infer directly information about network structure.

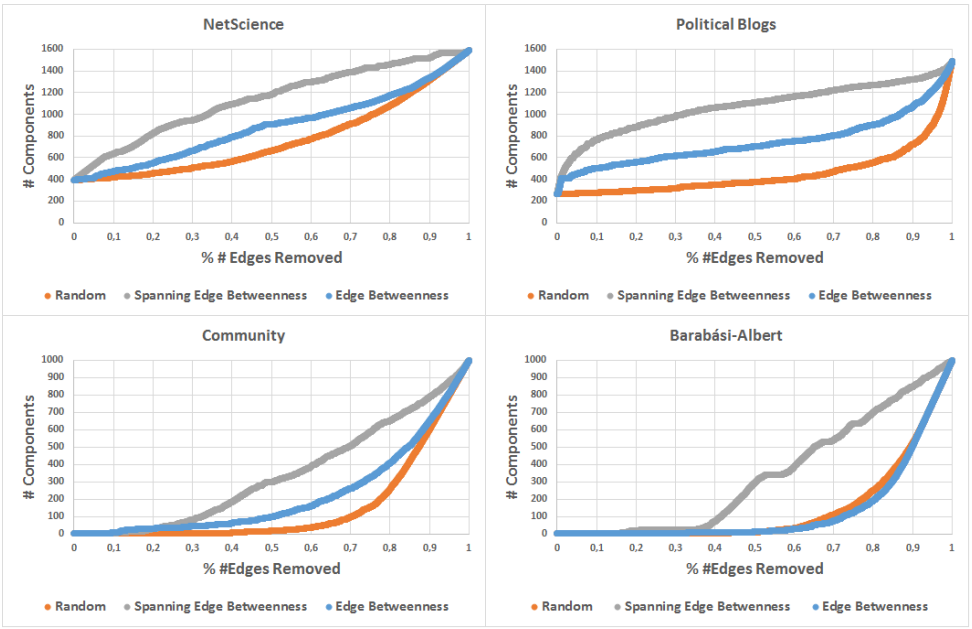


Figure 3.7: Analysis of Removing Edges. Three different criteria in NetScience, PoliticalBlogs, Barabási-Albert and Community networks: randomly, in decreasing order of spanning edge betweenness and edge betweenness values. We are able to observe that for all networks, empirical and randomly generated networks, removing edges in decreasing order of spanning edge betweenness leads to an earlier decomposition of each network when comparing with the other two methods.

Then, for each network mentioned before, we calculate the values of each measure – spanning edge betweenness and edge betweenness – for each edge and we sort the edges by decreasing order. The next step is to remove edge by edge on that order. For each edge percolation we register the number of connected components after each edge removal.

The results are as expected: removing by decreasing order of spanning edge betweenness speeds up the disruption of the networks when comparing with decreasing order of edge betweenness or randomly. In Figure 3.7, we show four examples – two from real networks and two from generated networks – but for all networks the results were similar on what concerns the number of connected components growth. For the same proportion of edges removed, removing edges with decreasing order of spanning edge betweenness breaks the network structure into more components than with decreasing order of edge betweenness.

3.5 Discussion

We presented a new edge centrality metric, the spanning edge betweenness, defined as the fraction of MSTs containing a given edge. We provide also required results and methods to compute exactly this metric. Since we rely on the Kirchhoff's matrix tree theorem, thus needing to compute several determinants for slight different matrices, we plan to investigate how to accelerate these computations by reusing previous computations and by using more efficient methods for sparse positive semi-definite matrices decomposition, such as those based on Cholesky's decomposition [144].

More recently spanning edge betweenness has been object of further studies. An improvement in what concerns the efficient computation of spanning edge betweenness was presented by Mavroforakis *et al.* [126]. The authors proposed a fast polynomial randomized approximation schema (FPRAS), relating spanning edge betweenness with the effective resistance measure [57] in electrical circuits. Their approach addresses unweighted graphs, and weighted graphs where weights denote edge mutiplicity, not being directly applicable on general weighted graphs. Hence, it is not trivial to extend it to our use case.

Qi *et al.* [145] introduced the concept of *spanning tree centrality*, that applies the same principles of spanning edge betweenness although applied to the nodes in a weighted network.

Biswas *et al.* [25] proposed a community-based link prediction, for identifying missing links or the links that are likely to appear in near future, using and comparing spanning edge betweenness, edge betweenness and k-path centrality. And, recently, Li *et al.* [119] proposed a new centrality measure, the *Kirchoff index*.

Spanning edge betweenness is a useful measure that can be applied both in weighted and unweighed graphs, allowing different types of evaluations – from confidence in phylogenetic trees to the

identification of edges that are critical to keep the network connected, passing through the ones that express redundancy and alternative network configurations.

In the phylogeny context, this work highlights the impossibility of selecting a MST based on the statistical support of the edges, and reinforce the importance of the biological plausibility of the model underlying the criteria for edge selection in presenting the best possible MST based proposal for the phylogenetic relationship of the entities under analysis.

In the network connectivity robustness context, we compared it with other measures, namely with traditional edge betweenness, and on several real and synthetic networks, concluding that spanning edge betweenness performs better at identifying the relevance of edges for maintaining network robustness.

4

Structural Balance

Contents

Network Science [15, 19, 54] has provided key insights on how individual states, from individual's choices [37, 38, 143], epidemic states [140], strategic behaviors [147, 158, 159, 164, 172], and opinions [36], among other traits, are locally influenced by their social ties and by the overall topology of interactions within a population. While the dynamics at the level of nodes is crucial, analogous dynamics occurs at the level of states and weights of links [14,114,115,136], with particular relevance within social settings.

Signed networks are networks where the links have a sign expressing some positive or negative ties between entities [35, 46, 64, 70, 92, 100, 114, 115]. With signed networks we can model social networks, where links represent friendship/enmity relations; financial networks, where links represent positive/negative correlations between firms; biological molecules, where link represent activation/inhibition between enzymes, proteins or genes; among others [10].

One of the key concepts used to characterize a signed network is structural balance. Structural balance is a concept developed by Heider [15], later adapted to a graph-theoretic model by Cartwright and Harary, that states that in a signed network, cycles containing an odd number of negative edges are a source of tension between the entities represented by the nodes [92]. This concept is particularly interesting when we think about our daily relations and even how some decisions can shape and influence other behaviors. Moreover, as we discuss in this chapter, structural balance characterizes *motifs* of interest in many real-world settings, from social to financial networks. Interestingly, the origins of such self-organized patterns of nodes and signs remains, to a large extent, a challenging open question.

Thus, in this chapter, we abandon the idea of looking only to edges or nodes to observe how these local patterns of interaction can impact the global structure of the network. In the next section we will provide more information about structural balance theory, presenting the global measure that we use throughout this chapter to analyze it, the *Degree of Balance* [35]. Next, we present three case studies in which we apply it.

The origin and self-organization of balanced patterns of interaction remain largely elusive. In our first case study (Section 4.2), we show how peer-influence, i.e., the fact that third parties may sway our perception of others, may lead to the emergence of the patterns of social balance observed in nature. To do so, we propose a new network model in which we allow edges to change their sign over time.

In Section 4.3, we analyze how social dilemmas may shape the corresponding signed network, built from the relation between cooperators and defectors in cooperation dilemmas. Our results suggest that, besides peer-influence, social balance may also emerge from ties of exchange and cooperation which prevail in our daily life.

In Section 4.4, we use financial networks to build signed networks with the goal to evaluate if there is a relation between financial volatility and the frequency of specific labeled motifs, obtained from the structural balance theory. We show that there is a a close relation between volatility and the number of

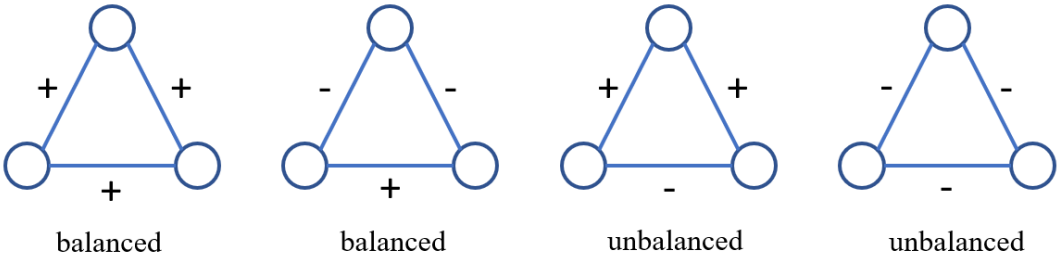


Figure 4.1: Social Balance Theory, by Cartwright and Harary [92]. The triads are considered balanced if the product of the signs are positive. Davis introduced the weak balance structure that considers all triads but the third to be balanced.

balanced positive triads.

Finally, from an algorithmic perspective, assessing the global degree of balance of a network implies counting all cycles of the network, leading to a well-known NP-hard problem [102]. This computational obstacle does not constitute the central research question addressed here, yet the difficulties associated remain subjacent in each of the case studies mentioned above.

4.1 Structural Balance Theory

In 1946, Fritz Heider published an initial study about how affective ties – as to like, to love, to esteem, etc., and their opposites – would influence interpersonal relations [15]. These simple cognitive configurations between people and objects led to the conclusion that a triadic relation is balanced if the three links are positive, or if two are negative and one positive, otherwise tension would emerge. The representation of balanced and unbalanced triads are illustrated in Figure 4.1. This was a first approach to social balance.

Later, Cartwright and Harary, extended this notion of balance to a graph – *structural social balance* – modeling a signed graph, where edges have positive or negative signs reflecting positive or negative ties between the individuals [35, 92]. They also extended the concept of balanced triad to balanced cycle, allowing cycles with more than three edges. A cycle is considered balanced if the product of the signs of its edges positive, i.e., if there are no odd number of negative edges in a cycle.

To measure structural balance they introduced the concept of *degree of balance* of a signed network as the ratio of the number of positive cycles to the total number of cycles. Let G be a signed graph, $c(G)$ be the number of cycles of G, $c_+(G)$ be the number of positive cycles of G, and $b(G)$ be the degree of balance of G. Then:

$$b(G) = \frac{c_+(G)}{c(G)}. \tag{4.1}$$

Later, Harary [93] also presented the *line index of balance*, which is a measure that analyze the number of links that must be changed/removed in order to achieve balance. This measure is also referred in the literature as the *frustration index* [9, 10, 70].

Following the work of Cartwright and Harary, in 1967, Davis [46] studied the relation between clustering and structural balance in graphs. The main question was about what conditions were necessary and sufficient for the graph to be separated into two or more subsets of nodes, where each positive edge would link two nodes of the same subset and a negative edge would link nodes from different subsets. The conditions was: a signed network is clusterable if and only if the network does not contain any cycle with exactly one negative link. This introduced the notion of *weak balance theory* as it allows for cycles/-triads to have all signs negative, allowing more than two subsets to be created. The main conclusion was that all balanced graphs are clusterable.

In the last decade, structural balance has received considerable attention. Doreian *et al.* [100] created an agent-based simulation model based on two levels: a micro-level that explores Heider's theory at an individual level, to minimize individual tension; a macro-level that explores Cartwright and Harary's at a group level dynamics. This simulation model is only for small groups dynamics as the designed variables have complicated impacts.

Facchetti *et al.* [70] implemented an algorithm for ground-state calculation in large-scale Ising spin glasses, to compute the global level of balance in large undirected networks.

Estrada and Benzi [64, 65] published a study presenting a walk-based measure of balance in signed networks, stating that contrary to what is generally believed real networks can be poorly balanced. They also show how unbalanced states can be changed by tuning the weights of the social interactions among the agents in the network.

Recently Giscard *et al.* [87] developed methods to evaluate balance on social networks from their cycles (of different sizes) and Aref and Wilson [10] formalized the concept of a measure of partial balance, discussing various measures and comparing them on synthetic datasets to investigate their axiomatic properties.

In our work, since our goal is to analyze not only the structural balance of a signed network, but also to analyze the proportion of each possible triad, we apply the degree of balance measure taking into account triads, that is, cycles of length 3.

4.1.1 On Counting Cycles

As presented in the previous section, obtaining the degree of balance of a network implies to find and count *simple cycles*, i.e., cycles that visit each node exactly once. This is the same that counting all the Hamiltonian cycles of the network, which is a NP-hard problem [102]. Due to its intractability, the degree of balance measure it is rarely used. Nonetheless, counting and enumerating cycles of different sizes is

a problem that has been continuously studied among time [5, 87, 108].

In our approach, we use the degree of balance measure using only cycles of size three. This is less demanding than to find all k-cycles for $k > 3$. Counting and finding triangles are commonly used in network analysis, p.e., when calculating clustering coefficient, and can be done in polynomial time [116].

Finding triangles is also special case of finding a subgraph or motif, problem that is related with subgraph isomorfism, which is also known to be NP-Complete. Nonetheless, recently Ribeiro *et al.* presented the g-trie data structure, specifically designed for discovering subgraph frequencies and which outperform the previous best methods for the same purpose [148, 149].

In the following case studies, we used both Leskovec *et al.* [116], and Ribeiro *et al.* [149] algorithms to build our model and conduct our analysis. We use the first for the peer influence model (Section 4.2) and the second on the analysis performed in Section 4.3 and Section 4.4.

4.2 The Origins of Social Balance and the Power of Peer Influence

4.2.1 Motivation

It is well-known that in social networks one can be friendly or unfriendly with others and that this can change over time. Moreover, individuals also shape and reshape their social environment themselves and are responsible for the specific features that characterize their social network [90, 111, 157, 164]. In this context, social media often reveals a complex interplay between positive and negative ties. Yet, the origin of such complex patterns of interaction remains largely elusive. In this chapter we study how third parties may sway our perception of others.

The works of Heider and Cartwright and Harary on structural balance theory state that in a triad, the relations of friend-enemy tend to converge to two balanced states: “the friend of my friend is my friend” and “the enemy of my enemy is my friend”, otherwise there will be tension between them [15, 35, 92]. Given this, our purpose in this case study is to model how the relations between individuals can change over time and if those changes converge to a balanced structure.

Our model is based on the relations with common friends, allowing peer influence and taking into account structural balance, following two simple ideas:

- “I will follow my friend’s beliefs”.
- ”I will follow the opposite of my enemies”.

Given this, we present a simulation model that, at each time step, evaluates if the sign between two individuals must change, based on those two ideas. Our goal is to observe if we are able to minimize tension across mutual friends (triads). As mentioned before, a triad is considered balanced if its edges

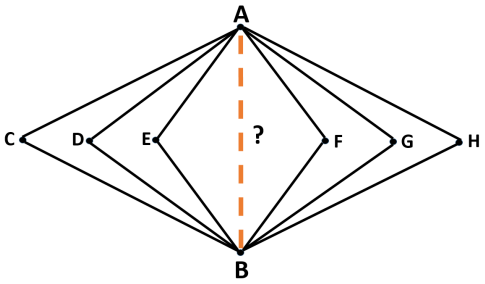


Figure 4.2: What will be the sign between A and B? It will depend on the majority of the signs of the products of each vertex A and B with each neighbour.

have the signs {+,+,+} or {+,-,-}, meaning that the product of its signs has to be positive (see Figure 4.1). We consider that the polarity of the relations is reciprocal, considering only undirected networks.

The results show that this simple model of peer influence promotes the signs between individuals involved in triads to converge towards an increase of structural balance.

4.2.2 Methods

Model of Peer Influence

Let $G = (V, E)$ be an undirected and signed graph, with $|V|$ vertices (individuals) and $|E|$ edges (ties), and with signed edges between two individuals a, b converted into weights: $w(a, b) = w(b, a) = 1$, if it is a positive tie, $w(a, b) = w(b, a) = -1$ if it is a negative tie. For each pair of individuals with friends in common, our model will count how many of those relations contribute with a positive and how many contribute with a negative sign.

Looking into the example illustrated in Figure 4.2: given a network let us consider individuals A and B that have C, D, E, F, G, H as friends in common. Following the two rules presented before – "I will follow my friend's beliefs" and "I will follow the opposite of my enemies" – we have to evaluate the values of $w(A, C)$ and $w(C, B)$:

- if $w(A, C) = 1$ the contribution will correspond to the same value of $w(C, B)$ - *I will follow my friend's beliefs*.
- if $w(A, C) = -1$ the contribution will correspond to the opposite of $w(C, B)$ - *I will follow the opposite of my enemies*.

In practice, we can summarize the calculus of each contribution as the product between $w(A, C)$ and $w(C, B)$. After doing the same for the other neighbours, we can now evaluate if we want to change the sign between A and B.

Algorithm 1: Update process for the sign of the edges taking into account triadic relations and peer influence.

```
for each user u do
    for each neighbour n do
        collect the friends in common
        for each friend in common c do
            if the product between w(u,c) and w(n,c) == 1 then
                pos(u,n) ⟵ pos(u,n) + 1;
            else
                neg(u,n) ⟵ neg(u,n) + 1;
            end
        end
    end
end
for each edge (a,b) do
    if pos(a,b) == neg(a,b) then
        there is no update and w(a,b) stays the same
    end
    if pos(a,b) > neg(a,b) then
        w(a,b) ⟵ 1
    else
        w(a,b) ⟵ −1
    end
end
```

Since we are interested in reducing tension among triads, the sign between individuals A and B will depend on a majority count between positive and negative products of the other relations in the triads related to that link.

Illustrating a little bit more: if $w(A,C) = -1$ and $w(C,B) = -1$, the product is equal to 1, so if we want the triad to be balanced we count this as a positive contribution, i.e., if the sign only depended on this triad it would be positive. If $w(A,C) = -1$ and $w(C,B) = 1$, the product is equal to -1, so if we want the triad to be balanced $w(A,B)$ would have a negative contribution in the count. If the sign only depended on this triad $w(A,B)$ would be negative. We remind that a triad is balanced if the product of the signs of its edges is positive. We note that each of this steps respects the two main ideas of peer influence that we presented earlier.

Simulations

We run two types of simulations, synchronous and asynchronous. The synchronous algorithm runs in two parts, as follows in Algorithm 1. In the end of each iteration – an iteration corresponds to the execution of both parts – we count the proportion of each four possible triads and calculate the degree of balance of the network. The simulations run until there are no more changes in the edge signs or until it reaches a given threshold on the counts changes. We stop the simulation when either the average of

Network	# Nodes	# Edges	% Edges +	% Edges -	# Triangles
HighlandTribes	16	58	50.00	50.00	68
Epinions	131 828	708 507	83.25	16.74	4770102
Slashdot	82 144	498 532	76.41	23.59	571127

Table 4.1: Statistics about the networks used in the simulations.

the fraction of the edges signs changed in the last two iterations is below 10^{-2}, or its difference for the last three iterations is below 10^{-4}. These thresholds were determined experimentally.

In the asynchronous model, after the count of positive and negative contributions we update immediately the sign of the edge being processed. The simulations with both synchronous and asynchronous algorithm lead to the same final results.

4.2.3 Experimental Analysis

We run our simulations in real networks with the original distribution of signs of each chosen dataset, but also with a random distribution of the signs, both in the same proportion as in the original network and in different proportions of positive and negative links. We also run simulations for well-mixed populations (cliques) using these networks as baseline.

In these experiments we used well-known signed social networks: Highland Tribes, the signed social network of tribes of the Gahuku–Gama alliance structure of the Eastern Central Highlands of New Guinea, from Kenneth Read (1954). The network contains sixteen tribes connected by friendship and enmity[1]; Epinions, a who-trust-whom online social network of a general consumer review site Epinions.com[2]; and Slashdot, a website which allows users to tag each other as friends or foes[3]. We also created cliques with different sizes just to compare complete connected networks with Epinions and Slashdot that are large-scale sparse networks. Because Epinions and Slashdot datasets are directed networks, we performed some operations in these networks to make them undirected. We analyzed each network and if some relation had a conflict - one edge in one direction positive, and in the other direction negative - we removed that edge, keeping only the relations that are reciprocal.

In Table 4.1 we can find the characteristics of each network. We processed each social network in three different ways: (1) we started by running the simulations with the networks as they were after removing conflicting edges; (2) for each network we randomly distributed the signs of the edges in the same proportion as in the original network; (3) for each network we distributed randomly and evenly positive and negative signs, i.e., 50% of positive edges and 50% of negative.

In Figure 4.3 we present the results of the simulations. It contains the initial and final distribution of the four possible triads and of the degree balance. As we can observe, the initial distribution of triads

[1] http://konect.uni-koblenz.de/networks/ucidata-gama
[2] https://snap.stanford.edu/data/soc-sign-epinions.html
[3] https://snap.stanford.edu/data/soc-sign-Slashdot090221.html

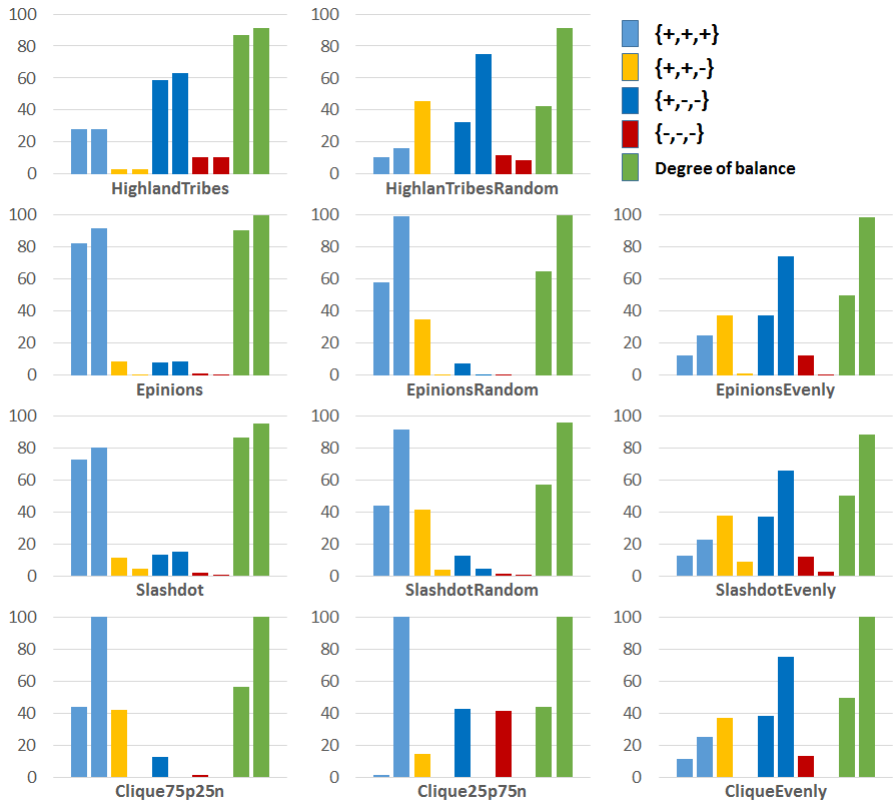


Figure 4.3: Simulations results. Each pair of columns corresponds to the initial and final distribution of each triad, except for the last pair which represents the initial and final degree of balanced of the network. As seen in Figure 1, only the first and third triads are considered balanced. *Random* means that the signs were distributed randomly in the same proportion of the original network and *Evenly* means that the signs were distributed randomly with $50\% - 50\%$ of positive-negative signs. HighlandTribes does not have Evenly because the distribution is already $50\% - 50\%$. We omitted the size of the clique, but we used sizes between 8 and 64 and the results were the same.

in the *random* and in the *evenly* networks are very different when compared to the original. Even when maintaining the initial proportion of positive and negative links, this means that there are some triads that are over represented in the original network, which indicates that the way signs are distributed initially has direct impact in the structural balance.

We can observe that having a dominant quantity of positive or negative links makes a network to converge to a dominant all-positive triads: (1) if individuals like each other and the friends in common also like each other, then there is no social reason to change; (2) if individuals do not like each other or the friends in common, then there are triads in tension and the update rule forces signs of all-negative triangles to change to positive. In all networks there is an increase of balanced triads, but depending on the initial distribution of signs, the dominant triads are different.

All networks with $50\% - 50\%$ of positive and negative signs, instead of converging to the same initial dominant triads, converge to the two-negative one-positive triads. There are two reasons for this to

happen: we have a high initial distribution of the triad $\{+,+,-\}$ that to decrease tension must change to $\{+,-,-\}$; the decrease of the distribution of the triad $\{+,+,+\}$, when comparing to the original networks, also indicates that there is not enough all-positive triads to compete with the new dominant $\{+,-,-\}$. This leads to the conclusion that initial distribution of positive and negative ties has direct impact in how signs can evolve and, again, in the degree of balance.

Making the signs evolve based on the balance theory criteria will always force the individuals of the network to act towards a minimization of tension. There is a strong tendency towards balance, but not always enough to achieve 100% balance. This happens in the non-fully connected networks, usually when the changes reach the threshold on the number of changes. We observe that the achievement of total degree of balance may depend on the connectivity of the network – fully connected networks eventually converge as can be seen in cliques. This last conclusion was already derived theoreticly in previous works by Antal [8] and Arnout van de Rijt [150]. With different approaches, both come to conclusion that in a complete connected networks, when updating triads with the goal of minimizing imbalance, a balanced state is achieved.

4.3 Structural Balance and Social Dilemmas

4.3.1 Motivation

In the previous section, we have shown how simple peer-influence principles can lead to high levels of social balance in networked populations. Social ties, however, besides defining patterns of acquaintances also encode links of exchange, cooperation, coordination, and trust, i.e., a broad range of strategic interactions which may strongly influence the sign and value of each social tie. In this section, we try to include part of this complexity in our understanding of the origins of social balance. We depart from most well-known 2-player dilemmas of cooperation to analyze the co-evolution of pro-social behaviours and balanced triads.

The self-organization and maintenance of cooperation remain a key open challenge in theoretical ecology and human evolution. Since Darwin, the interdisciplinary nature of this question has passionated generations of philosophers, biologists, mathematicians, computer scientists, or primatologists [12, 50, 58, 59, 98, 132, 162]. Cooperation, which may be loosely defined as an act evolving a cost which will benefit others, comprise a conflict between individual and collective interest: if everyone cooperates, all will be better off; however, due to the existence of a cost, individuals are often lead to defect. Such doomsday scenario of widespread defection is often referred to as the tragedy of the commons [94].

In the last decades, several mechanisms have been identified as active promoters of cooperation in natural and social settings [132, 146, 162]. Such a quest has profoundly influenced our understanding of natural selection and how we perceive the widespread levels of cooperation found in nature. Moreover,

such type of research may also be helpful to understand how cooperation may be fostered in novel situations in which cooperation is still absent, such as the maintenance of biodiversity [117], mitigation of the dangerous effects of climate change [20, 156, 186], or the overuse of antibiotics [184], to mention a few.

Cooperation has been traditionally studied in the realm of game theory, likely the most commonly used approach to formally describe conflicts of interest. The simplest social dilemmas [47, 124, 158] may be characterized as a game where two agents play with each other, choosing if they cooperate or not (defect), receiving an outcome (payoff) that can be optimal or not. The two choices result in four possible outcomes: R (reward for mutual cooperation), if both cooperate, P (punishment for mutual defection), if both defect, S (sucker's payoff, when a cooperator is cheated by a defector) and T (the temptation to defect, the payoff received by a defector when deceiving a cooperator. In this work we study three social dilemmas that depend on the ordering of the these payoffs [124, 158]: the Snowdrift-Game, $T > R > S > P$, the Stag-Hunt game, $R > T > P > S$, and the Prisoner's Dilemma, $T > R > P > S$, each representing a different social tension.

Whenever large populations of interacting agents are considered, most research has relied on evolutionary game theory [98, 132], the dynamical (and population-based) counterpart of classical game theory. Evolutionary games have been traditionally dealt with in unstructured populations, in which each agent interacts with all other agents.This setup can be conveniently described through the so-called replicator equation [98, 162], a deterministic equation which allows for the study of a fitness-based evolution in time. Evolutionary game theory may conveniently define both genetic evolution or a process of social learning in which, in the first case, individuals with higher fitness will reproduce more or, in the later, individuals with higher fitness will tend to be imitated more often. In any case, strategies that do better than average will grow, whereas those that do worse than average will diminish. Fitness is here defined as the average return each agent gets from interacting with all the other members of the population.

In most real-world situations, individuals are constrained to interact with (and imitate) a subset of the population, an idea conveniently defined as a network: Each agent is represented by a node that is constrained to play solely with its closest neighbours. The impact of topological constraints is known to induce profound evolutionary effects, as demonstrated experimentally in the study of the evolution of different strains of Escherichia coli [105]. In social settings, computational and mathematical models have also shown that cooperation is favoured on spatially structured population [133]. This result has been recently demonstrated experimentally with humans [120, 147]. Whenever some individuals engage in more interactions than others, individuals start facing broad distribution of fitness values [159]. Scale-free interaction structures, e.g., were shown to help cooperation to thrive [157–159] when compared with homogeneous interaction structures, as highly-connected nodes are promptly taken over by cooperators

who can then influence the whole community into cooperating. However, it remains an open question how cooperation can co-evolve and influence the degree of social balance (and vice-versa), the central question addressed in this section.

The evolution of cooperation in arbitrary network topologies brings additional complexities [142] which preclude a general analytical approaches, either by means of differential equations (for details see, e.g., [98]), or through large-scale Markov Processes [187]. Therefore, as in [158], here we shall rely on Monte-Carlo simulations of the evolution of strategic behaviors on complex networks.

Finally, having implemented such large-scale simulation setups, one can easily extend it to other types of social dynamics. Evolutionary game theory traditionally assumes that individuals revise their choices through a simple form of a social learning process, i.e., individuals revise their strategies by imitating those that are perceived as better. Here, we shall also analyze other types of strategic revision based on individual experience, employing a multi-agent reinforcement learning simulations on networks [185]. This will allow one to identify the role of each learning method, both on the emergence of cooperation and social balance.

Social Dilemmas with Social and Individual Learning

As already said, real networks have specific properties that can influence/determine the behaviour of the network [155, 158]. As we have seen in Chapter 2, real populations have been shown to be heterogeneous [15], in which some individuals have many more contacts than others. This contrast made it necessary to study the evolution of cooperation in complex networks, for which heterogeneity is large with degree distributions exhibiting a power-law behaviour.

In this case study we address the emergence of cooperation, in terms of the three social dilemmas referred previously, in complex networks with the purpose to study how these dynamics can shape the social balance of each network. We use two approaches to solve the dilemmas: social learning (through evolutionary game theoretical methods) and individual learning. Cooperation has been conveniently formulated in the framework of evolutionary game theory (EGT), which merges game theory and evolutionary dynamics. In this framework, each agent has a strategy that is updated after playing with its neighbours and imitating their behaviour (if perceived as successful). The payoff that each agent receives is then translated into fitness and the strategy is updated based on that fitness, with the goal of increasing it. The fact that strategies change over time makes the higher fitness strategy propagate over the network. Individual learning has also been used to study the evolution of cooperation over time. In this domain, agents learn from their interactions, relying on trial-and-error rather than imitation, with other agents. Each agent has a probability to cooperate that is updated using a reinforcement learning (RL) approach [171, 185]. Agents that learn in this way are essentially finite-action learning automata with a reward-inaction update scheme [130].

These two approaches differ in some important aspects. In EGT each agent must have knowledge about the strategy and fitness of all direct neighbours – the agent plays with all the neighbours, calculates the fitness and then changes its strategy if imitating a neighbour strategy (social learning) increases their fitness. In RL each agent simply resorts to its own experience in order to adapt behaviour over time, thus requiring less information to learn. It plays with only one neighbour at each time step, using the knowledge about the strategy used and the obtained payoff. This payoff is used to increase or decrease its propensity/probability to cooperate. While with EGT we have a distribution of the strategies over the population with RL we have a propensity to cooperate.

Based on the works of F. C. Santos, *et al.* [54] and S. Van Segbroeck, *et al.* [37], we study both social and individual learning models and show that the final degree of balance of each network follows the tendency for cooperation: in social learning increasing heterogeneity favours the emergence of cooperation, and thus the emergence of social balance; while with individual learning, the topology of the network does not have any significant impact on the propensity to cooperate, maintaining social balance, in most cases, in the same ratio as cooperation.

4.3.2 Methods

In this study we model social dilemmas as games and then use both social and individual learning to solve them in homogeneous and heterogeneous networks. In the end of each simulation, we create the corresponding signed network and calculate the degree of balance. In the next subsections we give the details of each step.

Social Dilemmas

We model interactions among individuals as symmetric two-player games in which both players can either cooperate or defect when interacting with each other.

$$\begin{array}{c} \\ C \\ D \end{array} \begin{array}{c} \begin{array}{cc} C & D \end{array} \\ \begin{pmatrix} R,R & S,T \\ T,S & P,P \end{pmatrix} \end{array}$$

Mutual cooperation leads to the reward, R (without loss of generality, we make $R = 1$), whereas mutual defection leads to the punishment, P (we make $P = 0$, thereby normalizing the advantage of mutual cooperation over mutual defection to 1 in all games). The other two possibilities occur when one player cooperates and the other defects, for which the associated game payoffs are S (sucker's payoff) and T (temptation) for the cooperator and the defector, respectively. Formally, these dilemmas span a four-dimensional parameter space. By normalizing mutual cooperation R to 1 and mutual defection P to 0, we are left with two parameters, T and S. We study the behaviour of all three dilemmas we mentioned

Algorithm 2: Strategy update based on social learning.

```
for each agent u do
    for each neighbour v do
        play a single round of the game and accumulate payoff Pu
    end
end
for each agent u do
    choose a neighbour v, randomly, among all Ku neighbours and compare Pu and Pv
    if Pv > Pu then
        update strategy of u (Su) with the probability given by (Pv − Pu)/[K> D>]
    end
end
```

before, in the ranges $0 \leq T \leq 2$ and $-1 \leq S \leq 1$, which is sufficient to characterize the games under study.

Evolutionary Dynamics - Social Learning

Evolution is implemented based on the work of Santos *et al.* [54] with the finite population analogue of replicator dynamics [85, 96]: in each time step (generation), each agent engages in single rounds of a game with each directly connected neighbour, and accumulate the payoffs. At the end of the generation, all agents update their strategy simultaneously with some probability. Algorithm 2 shows the complete algorithm. Where $K_> = max(Ku, Kv)$ and $D_> = max(T, 1) - min(S, 0)$. Updating synchronously or asynchronously does not bring any qualitative modifications to the results presented. This is done until there is no significant changes on the distribution of the strategies.

Individual Learning

Individual learning is implemented based on the work of S. Van Segbroeck, *et al.* [37]. In each time step (iteration) agents may interact with each other only if they are directly connected in the network. Each agent learns from its interactions, adjusting its strategy accordingly. The strategy of an agent is encoded as a single time-dependent probability $p(t)$ to cooperate. Each agent uses the payoff it receives upon interaction to update its strategy. Algorithm 3 shows the complete algorithm. Equation 4.2 shows that when an agent chooses to cooperate, its probability to cooperate in the future will increase proportionally to the obtained feedback. Similarly, an agent who chooses to defect will become more likely to defect in the future. The parameter λ, known as the learning rate, specifies the immediate impact of the feedback on the agent's strategy. The parameter $\beta(t) \in [0, 1]$ denotes the feedback the agent obtains at time t. This feedback is given by the game payoff the agent received, divided by the maximum possible payoff value.

Algorithm 3: Strategy update based on individual learning.

for each agent u **do**
choose a randomly neighbour v
play a single round of the game
update $p_u(t+1)$ and $p_v(t+1)$ according to the rule

$$p(t+1) = \begin{cases} p(t) + \lambda\beta(t)[1-p(t)], & \text{when the agent chooses to Cooperate at time t} \\ p(t) - \lambda\beta(t)p(t), & \text{when the agent chooses to Defect at time t} \end{cases} \quad (4.2)$$

end

Building the Signed Network

To calculate the degree of balance of a network, the edges of the network must have signs. In this study, for this to be possible, we have to build the signed network inherent to each game. Each sign, between two agents, is distributed as follows:

- If both agents cooperate the sign is positive, representing a mutually satisfactory partnership.
- If both agents defect the sign is negative, as both agents are unsatisfied with this interaction.
- If one of the agents cooperate and the other not the sign can either be positive or negative with probability of 50%. We also tested assigning always a negative sign in this case, but the results were indistinguishable.

In the end we apply the degree of balance to triads, that is, cycles of size 3.

Simulations

We run both learning models in networks of fixed size $|N = |V| = 1000$. We used homogeneous and heterogeneous networks. For homogeneous networks we used fully connected networks, representing well-mixed populations, and regular networks, representing regular ring lattices, where each agent has the same number of neighbours and is connected to those that are closest [190]. For heterogeneous networks we used random scale-free networks, following the B-A model and the DMS model, with power-law degree distributions. As seen in Chapter 2, in the B-A model, the network grows adding a new node, at each time step, and connecting it to *m* other nodes already in the network. This connections are probabilistic, depending on the degree of the nodes to be connected with, making older nodes having higher degrees, creating hubs. This is the combination of the two processes – *growth* and *preferential attachment* [13]. In the DMS model, each time a node is added, instead of choosing other nodes to connect with, it chooses one edge randomly and connects to both ends of the edge [56]. Except for the fully connected network (where $Z = N - 1$), all the networks used in the experimental evaluation have average degree $Z = 8$.

For $R = 1$, $P = 0$, $0 < T < 2$ and $-1 < S < 1$, all simulations start with an equal percentage of strategies: in EGT 50% of Cooperators and Defectors, in RL each agent starts with a propensity to cooperate of 50%, both distributed randomly. Because this can influence the final results, we run 50 different runs for each possible combination of values in the range of R and S, calculating the average in the end. This way, all vertices are initially populated with a strategy, and no initial advantage is given to cooperators or to defectors. For all networks, the topology of the graph remains frozen throughout the simulations.

Equilibrium frequencies of strategy distribution (EGT), and propensity to cooperate (RL), were obtained by averaging 500 iterations after a transient time of 11000 iterations for EGT, and 125500 iterations for RL. Both number of iterations correspond to each convergence state. Furthermore, for each network, final data results were obtained by averaging over 50 simulations, corresponding to 50 different runs for each possible combination of values in the range of R and S.

In the end, we run the algorithm to distribute signs over the networks and calculate the final degree of balance.

4.3.3 Experimental Analysis

We study both evolution of cooperation and evolution of social balance as a function of two parameters: S, the disadvantage of a cooperator being defected (when $S < 0$), and T, the temptation to defect on a cooperator (when $T > 1$). We present the results in Figures 4.4, 4.5, 4.6, and 4.7. Each figure contains the results for both EGT and RL. In each heat map, the horizontal axis corresponds to the range of T's values, the vertical axis corresponds to the range of S's values. Also, each heatmap is divided in four quadrants representing a default non-threat scenario and the three social dilemmas:

- $S \geq 0$ and $T \leq 1$ – upper-left quadrant – Harmony Game, no threats.
- $S < 0$ and $T \leq 1$ – lower-left quadrant – Stag-Hunt domain (SH).
- $S \geq 0$ and $T > 1$ and $(T + S) > 2$ – upper-right quadrant – Snowdrift game domain (SG).
- $S < 0$ and $T > 1$ – lower-right quadrant – Prisoner's Dilemma domain (PD).

Each heat map in the first line corresponds to the result of applying EGT. The second line corresponds to the results obtained applying RL. First Column of each line corresponds to the percentage of cooperators (EGT) and to the propensity to cooperate (RL). Second Column is the Degree of Balance of the final network. When applied, the third column is the Degree of Balance of the signed network with the signs of the final network distributed randomly. We do this to analyze if the degree of balance would be significantly different from the final network, after solving each social dilemma, if we would have the same networks with the same proportion of positive and negative links, but now without any relation with

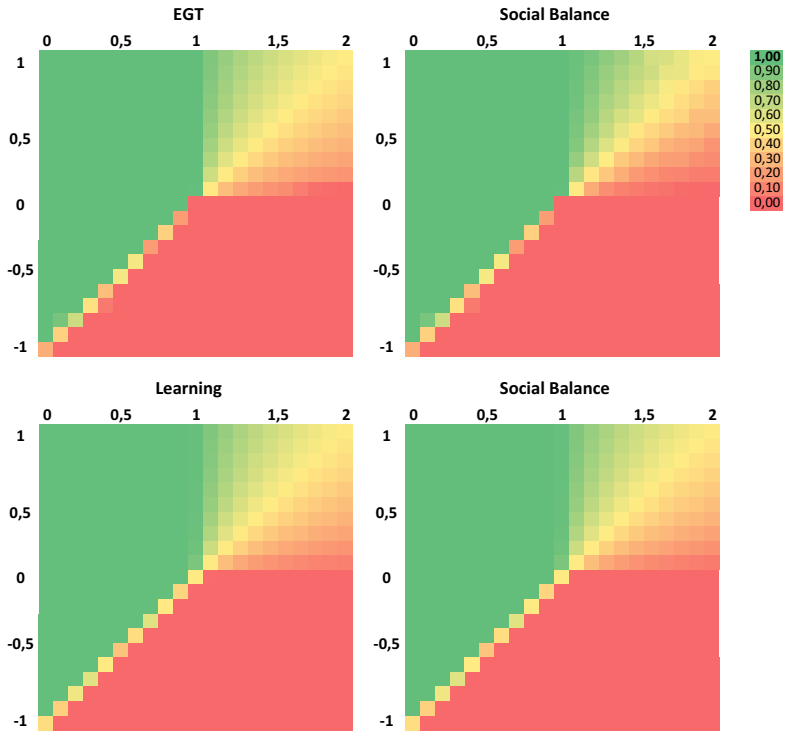


Figure 4.4: Results for fully connected networks. Evolutionary dynamics (upper panels) and reinforcement learning dynamics (lower panels) and resulting social balance in fully-connected networks for each case. For each case, stationary cooperation levels (left panels) and social balance (right panels) are plotted as a contour drawn as a function of two parameters: S (the disadvantage of a cooperator being defected) and T (the temptation to defect). In the absence of any of these threats ($S < 0$ and $T < 1$; upper-left quadrant) cooperators trivially dominate and social balance becomes prevalent. The lower-left quadrant ($S < 0$ and $T < 1$) corresponds to the Stag-Hunt domain (SH), where the population either ends coordinating into full cooperation or full defection. The upper right quadrant ($S < 0, T > 1$) corresponds to the Snowdrift game domain (SG), where, as expected, one observes a stable co-existence of cooperators and defectors. In all quadrants, the levels of social balance follows the prevalence of cooperators. These results provide the reference scenario with which the role of population structure will be subsequently assessed for other topologies (details provided in main text). Moreover, it suggests an equivalence between social and individual learning, for all classes of 2-player symmetric games, in the absence of an interaction structure.

social dilemmas. We will observe that the degree of balance is always higher in the final network after solving social dilemmas. The correspondence between scales and colors are given in each figure.

We start with the fully connected network (Figure 4.4), using it as a baseline, as it represents a well-mixed population. For both approaches, the results for cooperation are the expected, as already shown in [37, 54]. We find that the degree of balance is like a mirror of cooperation. In all dilemmas, the amount of cooperators will dictate how is the network balanced. The results for the random distribution of the signs is not presented as they did not show any difference from the final social balance.

As we increase heterogeneity, we begin to see some differences. As it was shown in previous works by Santos, *et al.* [54] and S. Van Segbroeck, *et al.* [37], in EGT, increasing heterogeneity increases the

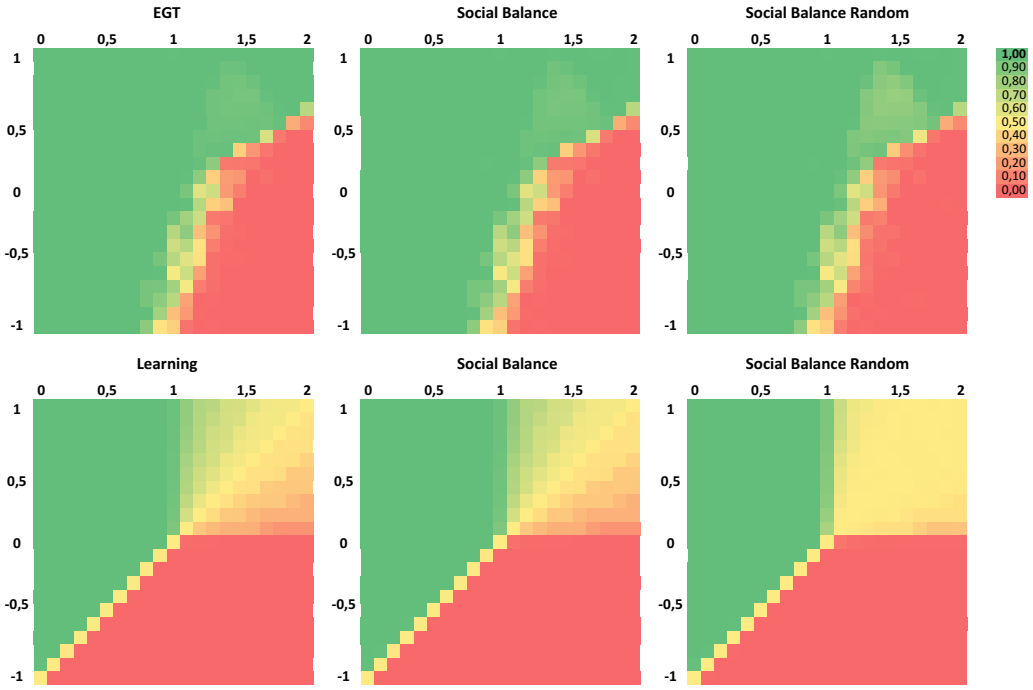


Figure 4.5: Results for regular ring lattices with average degree $Z = 8$. The panels follow the same logic as the previous one, but with a third column in which we present results for the social balance if the same proportion of positive and negatives signs was distributed randomly (details provided in main text).

amount of cooperation, but the same does not happen in RL (individual-based learning). The topology of the network may have a great impact in the spread of cooperation when solving social dilemmas with EGT, but with RL it does not have any significant change from the baseline with fully connected networks.

In what concerns social balance, the major impact can be observed in Figures 4.5, 4.6 and 4.7, where for the RL approach, in the Snowdrift game, upper-right quadrant, there is an increasing difference between the final degree of balance of each network, and the degree of balance of the same network if we randomly distribute the same proportion of positive and negative signs through the network.

We can also observe that in B-A (Figure 4.6) and DMS (Figure 4.7) models, there is a clear division between balanced and unbalanced states depending on the parameters of S and T, even if the propensity for cooperation is similar.

In this preliminary study we can take some conclusions related to the comparison between social and individual learning, to the relation between cooperation and social balance, and to the effect of increasing the cluster coefficient effect. When comparing social and individual learning, it is possible to conclude that individual-based learning is not influenced by the structure of the population (network), what was shown not to be true in real experiments [147]. This also suggests that human behaviour is more likely to be modeled through peer influence and social learning, as was shown in the previous Section and in

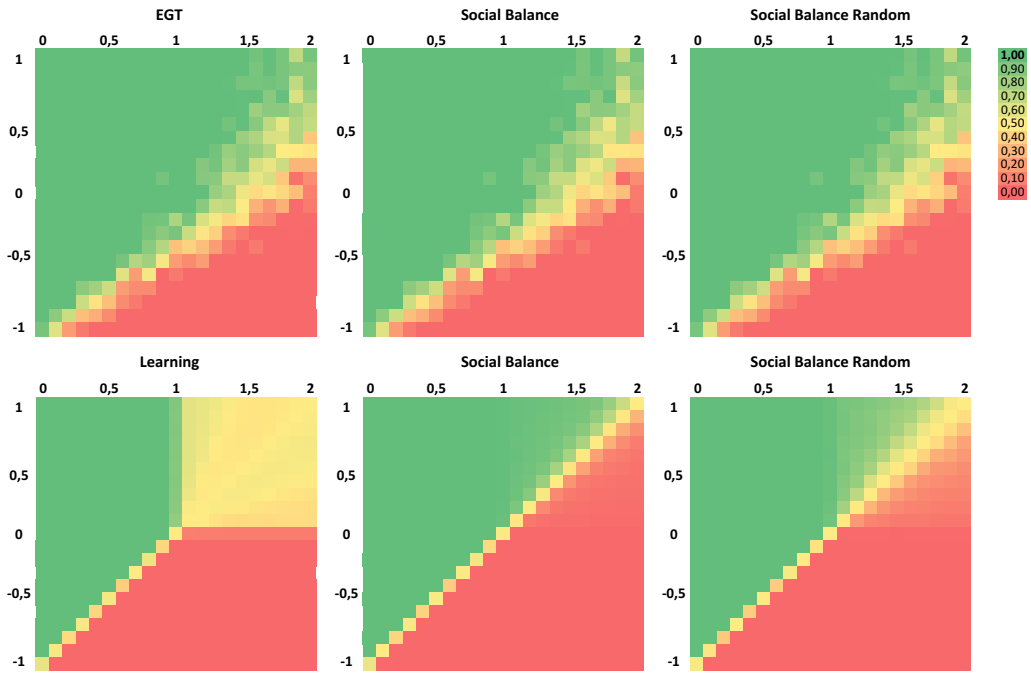


Figure 4.6: Results for B-A model networks with average degree $Z = 8$. The panels follow the same logic as the previous one (details provided in main text).

this simulations with social learning. Another conclusion is that, as we have stated before, the results suggest a direct relation between cooperation and social balance. This fact is reinforced when we focus our attention between the second and the third panels of each figure of the results and we observe that there is a less proportion of social balance if we randomly distribute the signs of the edges. Finally, we are also able to identify that when we increase the cluster coefficient of a network, from de B-A model to the DMS model, the number of defectors drop considerably.

4.4 Structural Balance and Volatility in Financial Networks

Measuring the inner characteristics of financial markets risks have been proven to be key at understanding what promotes financial instability and volatility swings. Advances in complex network analysis have shown the capability to characterize the specificities of financial networks, ranging from credit networks, volatility networks, and supply-chain networks, among other examples. Signed networks have been used in the past to study financial portfolios (see, e.g., [46]). One of the main observations is that usually a portfolio presents high values of structural balance, being rare to have unbalanced relations. Here we use signed networks and the information about the frequency of specific motifs (triads) to identify financial volatility. Throughout this chapter we extend the definition of motif to a subgraph, in this context

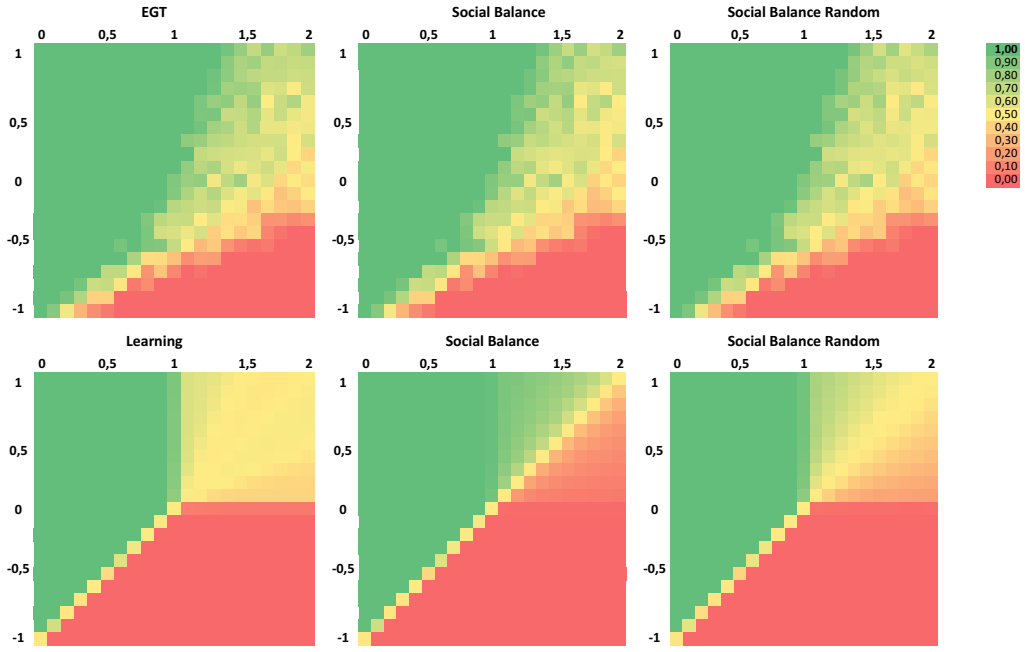


Figure 4.7: Results for the DMS model networks with average degree $Z = 8$. The panels follow the same logic as the previous one (details provided in main text).

a triad, that is fully signed - both edges and vertices have signs - and whose frequency on the network is higher than expected by random.

4.4.1 Motivation

In recent years, it is undeniable how interconnected financial markets are becoming [3, 31, 60, 113, 123]. Modern economic agents and their institutions can operate globally in a interdependent and connected market. Proof is the most recent financial crises, in which we observed the growing importance of accounting for systematic risk, where a single financial institution can affect the global market and all its agents [16–18, 21–24, 189]. As a result, financial systems are a natural playground for network science. It is both convenient and insightful to describe the various connections between financial assets and institutions in the form of a graph. In Allen et. al [3], it is detailed the vital role of network connections in the interbank market. Banks are exposed to their peers, both by holding crossed positions (mutual exposure in their balance sheets) as well as sharing similar market portfolio, assets and liabilities (as creditors and depositors) [3, 31, 113]. Those so-called markets represent market players exchanging cash-flows and goods between them, being responsible for managing one of the most complex multi-agent systems humanity have ever created, permanently connecting every corner of the globe [167,193].

In this context, it remains to a large extent an open question how one can extrapolate from network properties to commonly used measures to detect risk [1, 4, 16–18, 21–24, 89, 113, 189]. At a firm level,

as detailed by Onnela *et al.* [135], it is possible to assess firms' performance, looking at their stock price time series. It is believed that firms' valuation is based on all the available information [28, 71]. This concept of "pricing" is understood as the current fair value of all company assets and the present value of future expected income flows from the current asset allocation. In other words, it measures how much the firm is worth. Nonetheless, these companies are not isolated. They are interacting with one another by exchanging cash-flows, products, clients, among others [31, 122, 135]. Despite the fact that is not straightforward to measure the nature of all this inner relations, stock price variations are the closest to public information it can be used to study this complex system [135, 167].

Some prior studies tried to capture financial stability (or risk), looking to the inner characteristics of the network. Battiston and Caldarelli *et al.* gave us the insights of how it is possible to detect systemic risk looking at interbank network, in the context of asset-liabilities relations [16–18, 21–24, 189]. They studied the impact of several types of networks (from random to scale-free) and sources of information (the degree at which nodes can accurately measure the risks of his peers), to access the likelihood of a bankruptcy cascade effect in a distress financial period [18, 21–23, 189]. Moreover, these authors proposed some network measures, like the debt rank [16, 24], to better capture the importance of each single node in a financial network in a crisis context. Boss *et al.* [31] presented some encouraging results in the Austrian Interbank Market proving that network metrics can explain structural changes in the environment and are similar to some studies in other scientific fields. Nonetheless, not having the capability of seeing a time-varying picture and an exogenous indicator to validate the results, it was still difficult to proof that it was not a random coincidence. In Onnela *et al.* [135, 137], important steps were made when it was introduced the time dependence, but still misses the study of the co-movement between network characteristics and an external validation. However, by having the opportunity of using US financial data, namely the companies that are part of the most relevant Stock Market Index (S&P 500), it is possible to check if the network metrics mimic the volatility and financial systematic risk, by comparison to the "Fear Index", the so-called "VIX Index". This index measures the implied market risk and accounts for the short-term implied-volatility derived from all the S&P stocks and its options and structured products [68].

Our aim is to study the financial systematic risk through networks characteristics, resorting to both weighted and signed networks representations. Using as framework real US data, we build a network from price correlations among companies and measure the likelihood of capturing volatility shifts looking at the time variation in particular at specific motifs. The identification of specific motifs, rooted in the structural balance theory [35, 46, 92], allow us to perceive network shifts and their impact on network volatility. This way we are able to analyze the volatility of a network with specific local patterns. We find a statistical significance when it comes to explain and replicate what happens to VIX Index in a given period of time.

4.4.2 Methods

Financial Markets Structure

We build our financial networks based on correlation matrices, as introduced before. We consider a price time series from 1992-2018, whose source is Bloomberg database[4], for a security set of 500 companies. Let us define $P_i(\tau)$ as the closing price of stock i at time τ and the daily logarithmic return of stock i as:

$$r_i(\tau) = ln[P_i(\tau)] - ln[P_i(\tau - 1)]. \tag{4.3}$$

Then, by defining a time grouping window of bulk size *T*, one obtains a cube of correlations, where $\rho(x, y)_{[t,t+T]}$ is the returns correlation between firm x and firm y between t and $t + T$ (giving n data points), defined as:

$$\rho = \frac{\sum_{i=1}^{n}(x_i - \overline{x})(y_i - \overline{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \overline{x})^2(y_i - \overline{y})^2}}. \tag{4.4}$$

Having calculated the correlation cube a weighted time-varying network was built, whose nodes were the constituents of the S&P 500 Index and their links were the correlations throughout time. By this it is meant that whenever two firms have a correlations different from zero (and ranging between -1 and 1), they will share an edge whose value is their correlation. As an example, if firm A has a price correlation in a given set of days (lets say 0.75) with firm B, the network will display two independent nodes (A and B) whose edge between them has a linkage (during this time-frame) of 0.75. From here, we build a signed network, in which every pair of nodes that share a correlation below a given threshold will have a negative sign, whereas a positive sign is defined whenever the value is greater that the threshold. Let's consider a threshold with given value X. All links with values below $-X$ will have a negative sign and above X a positive sign. The links with values between $-X$ and X were not considered to avoid adding noise or spurious edge relation for very low correlations.

Relation between VIX and Structural Balance

Let $G = (V, E)$ be an undirected and signed network, with $|V|$ vertices (individuals) and $|E|$ edges (ties), and with edges labels $w = \{-1, 1\}$ between two assets (a,b): $w(a, b) = w(b, a) = 1$, if it is a positive correlation, $w(a, b) = w(b, a) = -1$ if it is a negative correlation. To calculate the degree of balance of a network we first use gtrieScanner [5] [149] to obtain all triads of the network. Then, for each triad, we calculate the product of its signs and in the end we obtain the degree of balance.

As was already observed by Harary [46] financial networks tend to have high values of structural balance. We are able to observe the same in our datasets, see Figure 4.8. Given this, and because

[4]Which is one of the most accurate database when we are dealing with financial data. The dataset has, on average, 252 days per year during 26 years.

[5]`http://www.dcc.fc.up.pt/gtries/`

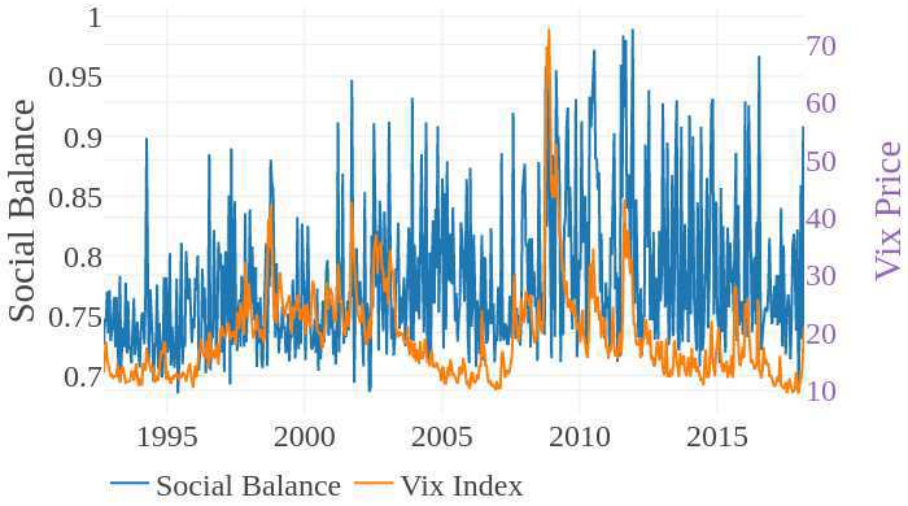


Figure 4.8: VIX Index Price and Social Balance Time Series (1992-2018).

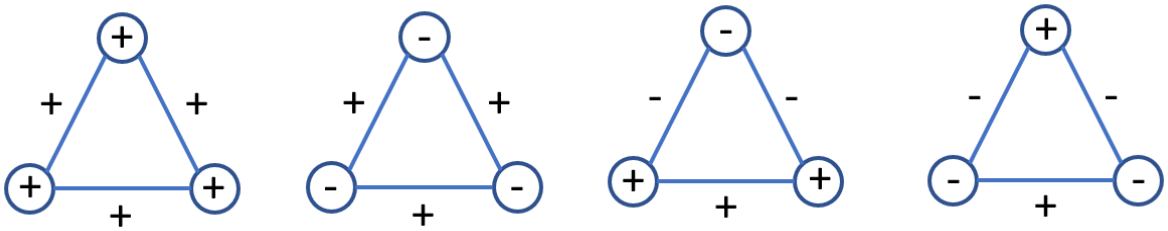


Figure 4.9: Balanced triads with signs on the nodes. Those signs correspond to the performance of the firms represented by the nodes. A positive sign (+) is inputed whenever for the given time-frame node value has gone up, whereas a negative sign (-) is given when a given node lost value for the same period.

when trying to relate to the VIX we want to extract the most discriminatory network characteristics effect, we removed the unbalanced triads, focusing only on those that are balanced. Additionally, since in our edge definition it was chosen firms' prices correlations we could not differentiate triads whose components were trending up versus those that were jointly falling apart, we added into each node a positive or negative sign respectively to their performance in the time horizon in study, looking for the frequency of specific motifs in the form of Figure 4.9.

Therefore, as we calculate the product of the signs of each triad, we also gather the frequency of each different motif. The goal is to analyze if some specific motif can relate to VIX, always maintaining the notion of structural balance in the signs of the edges and using the signs of the nodes as supplementary information. Within such context, it is now possible to split those balanced triads with a positive impact on the network from those whose performance was poorly, relative to its peers.

Number Of Days	Correlation Cut-off Threshold	Statistical Significance (Z-Scores)								
		Unbalanced	Balanced	Unbalanced	Balanced	Balanced				
		Triads (0 Pos Edges)	Triads (1 Pos Edges)	Triads (2 Pos Edges)	Triads (3 Pos Edges)	Triads (1 Pos Edges) 1/3 Pos Nodes	Triads (1 Pos Edge) 2/3 Pos Nodes	Triads (3 Pos Edges) 0 Pos Nodes	Triads (3 Pos Edges) 3 Pos Nodes	Social Balance
5	0.00	-1.179	-0.795	-0.247	1.015	-12.226	10.840	-15.727	16.167	0.685
5	0.40	-1.124	-1.069	-1.124	1.344	-10.279	9.611	-15.869	16.095	-0.302
5	0.80	-469.342	-0.685	-331.124	0.521	-8.945	7.007	-15.657	15.107	-53.078
10	0.00	-3.122	-2.885	-3.919	4.729	-9.028	5.138	-10.254	9.787	3.361
10	0.40	-3.122	-3.679	-1.709	3.521	-8.301	3.441	-9.966	9.787	1.943
10	0.80	-234.229	-3.838	-147.130	3.123	-5.964	1.164	-9.776	9.224	-28.268
15	0.00	-2.098	-3.561	0.000	2.486	-4.664	3.663	-5.595	5.602	0.855
15	0.40	-3.363	-2.873	0.000	2.777	-3.958	2.777	-5.490	5.497	-0.095
15	0.80	-126.727	-1.714	-76.267	1.236	-2.484	2.292	-4.972	4.567	-21.328
30	0.00	0.403	-0.672	0.000	0.807	-4.945	1.348	-5.262	5.442	-0.134
30	0.40	-4.789	-0.403	-0.806	0.403	-4.789	0.672	-5.103	5.442	0.134
30	0.80	-62.640	0.000	-47.842	-0.403	-2.308	1.485	-3.587	3.742	-12.086

Table 4.2: Sensitivity Analysis: Z-scores from Statistical Tests (Green values are Statistical Significant with 95% confidence).

Number Of Days	Correlation Cut-off Threshold	Accuracy Versus Vix Index								
		Unbalanced	Balanced	Unbalanced	Balanced	Balanced				
		Triads (0 Pos Edges)	Triads (1 Pos Edges)	Triads (2 Pos Edges)	Triads (3 Pos Edges)	Triads (1 Pos Edges) 1/3 Pos Nodes	Triads (1 Pos Edge) 2/3 Pos Nodes	Triads (3 Pos Edges) 0 Pos Nodes	Triads (3 Pos Edges) 3 Pos Nodes	Social Balance
5	0.00	48.38%	48.91%	49.66%	51.39%	34.11%	64.24%	30.20%	70.25%	50.94%
5	0.40	48.46%	48.53%	48.46%	51.84%	36.44%	62.73%	30.05%	70.17%	49.59%
5	0.80	0.15%	49.06%	0.30%	50.71%	38.09%	59.43%	30.28%	69.12%	8.79%
10	0.00	43.99%	44.44%	42.49%	59.01%	33.48%	59.76%	31.53%	67.72%	56.46%
10	0.40	43.99%	42.94%	46.70%	56.76%	34.68%	56.61%	31.98%	67.72%	53.75%
10	0.80	0.30%	42.64%	0.75%	56.01%	38.74%	52.25%	32.28%	66.82%	13.06%
15	0.00	45.05%	41.67%	50.00%	55.86%	39.19%	58.56%	37.16%	62.84%	52.03%
15	0.40	42.12%	43.24%	50.00%	56.53%	40.77%	56.53%	37.39%	62.61%	49.77%
15	0.80	0.68%	45.95%	1.80%	52.93%	44.14%	55.41%	38.51%	60.59%	14.41%
30	0.00	51.35%	47.75%	50.00%	52.70%	34.23%	54.50%	33.33%	67.12%	49.55%
30	0.40	34.68%	48.65%	47.30%	51.35%	34.68%	52.25%	33.78%	67.12%	50.45%
30	0.80	1.35%	50.00%	2.25%	48.65%	42.34%	54.95%	38.29%	62.16%	18.47%

Table 4.3: Summary of runs accuracies: number of times the variation in VIX Index was correctly replicated by the tested type of triads (in Percentage points).

4.4.3 Experimental Evaluation

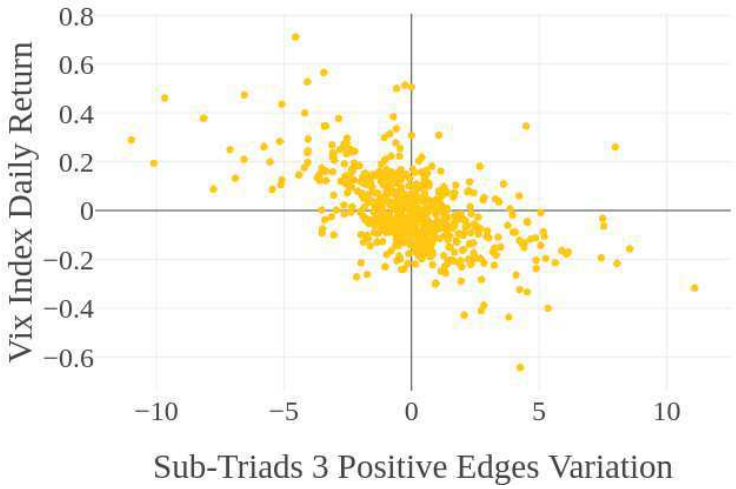


Figure 4.10: Correlation between VIX Index variation and balanced sub-Triad with 3 positive edges variation (with 11 Days and 0 correlation Cut-off Threshold, we got a replication power of 67.85%).

To answer our baseline question, which is the study of the interplay between structural balance and the degree of risk or volatility, we considered the data from the S&P constituents to input in the network previously explained, as well as the VIX Index, as our exogenous and independent variable.

We measure the impact that the different triads have throughout time when compared with the volatility index. Balanced triads tend to overweight the unbalanced ones, being the later even in lower number in the presence of large volatility swings. This outcome lead us to perform a set of 50 runs to test if the likelihood of balance and the different triads replicate the uncertainty of the market. From Table 4.2, we conclude that not only the correlation of triads with three positive edges with the VIX index is the only that it is statistically significant different and higher than 50% but also portray replication and/or mimic values consistently above 50% (as displayed in Table 4.3).

Due to fact that only triads of fully positive edges seem to matter when mimicking the "fear Index", we split those triads whose nodes were positive in a given time-frame, versus the ones that were negative in the same period of time. In Figure 4.10 we show that looking to the node performance, we might have gain some additional explanatory power. In Table 4.2 are the p-values from the sensitivity analysis run within the presented context. As a matter of fact, we obtained significantly higher results using nodes information for the considered period. Without having a correlation cut-off threshold and a grouping size-window of 5 trading days, motifs with positive edges and positive performance of their nodes, VIX index is replicated at a rate of 70%.

The present results are in line with our empirical reasoning that whenever there are times where all firms are moving together, is not because of their fundamental or intrinsic value, but mostly because of something exogenous in the financial world that spreads out into the entire network, despite its positive or negative impact. From this results it is possible to observe that when the VIX Index tend to spike, triads with three positive edges and nodes tend to shrink and increase otherwise, as described in Figure 4.10.

4.5 Discussion

The study of structural balance has benefited enormously from the quick growth of available data and more models/tools are needed to understand its particular dynamics. In this chapter we presented theory and methods to study structural balance in different types of networks.

In Section 4.2 we presented a study about the origins of structural social balance with roots on peer influence. We report the results of a simulation approach to understand how the relations between people can evolve taking into account peer influence (friends in common) and, given that, which is the impact in the evolution of social balance. The conclusion was clear, social balance can emerge from simple peer influence mechanisms.

In Section 4.3 we presented a preliminary study about the interplay between social dilemmas and

social balanced using the two most common approaches to solve social dilemmas: evolutionary game theory and reinforcement learning. We calculated the degree of balance for each dilemma in homogeneous and heterogeneous networks and the results show that social balance can also emerge from cooperation mechanisms. The results are particularly illuminating in the case of the Snowdrift Game (SG), a social dilemma that motivates the co-existence of Cooperators and Defectors. While in PD and SH we mainly obtain the full dominance of either Defection or Cooperation, leading to a trivial association with extremely low and high social balance (respectively), for SG the final balance of the networks depends on the resulting spatial distribution of C's and D's. Here we find that the final distribution of cooperators and defectors allows a perfect match between average cooperation and average social balance in the network, this way revealing the interesting connection between the two measures.

In Section 4.4 we study the relation between the frequency of specific motifs and the financial fear index – VIX. We modeled a financial network as a signed network, where signs correspond to a positive or negative correlation between firms, and then e collected the frequency of each structural balance triad. When we observed that the network was highly balanced, we extended the notion of balanced triads putting signs also on the nodes. This allowed us to identify which financial patterns (firms going up or down) were dominant. Our results show a close relation between the frequency of specific balanced motifs and VIX.

These three studies suggest that studying structural social balance has the potential to uncover much more about network dynamics through the observation of labeled local patterns.

5

Simulations of Networked Systems

Contents

Until now, in the previous chapters, we studied systems that can be modeled by a single network. However, it is rare that a system exists only by itself, isolated from others. For example, in epidemiology diseases can spread within populations but they can also be transmitted to other populations. If we think about communications networks, it is easy to realize that these networks will only be functional if power grids are active. In recent years, with the considerable advances in Network Science and modern technology, these so called networks of networks have gain a considerable amount of attention [33, 43, 52, 81, 101, 104, 188].

Networks of networks have been studied as multilayer networks and interdependent networks. Multilayer networks are networks that share the same entities among layers but where each layer correspond to a different system [26, 48, 109]. One simple example is to model the different social networks in which a person can be a part of. Interdependent networks are networks of networks that share some interdependence between them [151]. This interdependence can be on the structural and/or functional behaviour of the coupled systems [81, 188]. It has been observed that the traditional statistical observations in single networks are different from those in networks of networks [33]. This demands new models and tools to study systems represented as networks of networks.

Given the large-scale dimension of these systems, new computational challenges arise naturally. If large-scale single networks already demand powerful computational tools, working with networks of networks is even more challenging.

In this chapter we propose a large-scale simulation framework for bacterial populations over host contact networks. This work is motivated by the necessity of simulating real data related with the evolution of bacterial populations. In general, this limitation – the availability and quality of the real data for a given problem – is one of the main motivations to develop simulation frameworks. In the field of microbiology, one of the challenges is the necessity of validating models and methods applied in phylogenetic inference. Most publicly datasets used for phylogenetic inference are biased to their original studies and they are only small samples of the genetic material which may also affect our bias or perception of the true population structure.

We model our system as a network of networks where each node of the host contact network is, itself, a fully connected network, representing a population of bacteria. Inside these networks, each bacteria can interact with each other given a set of rules. The dynamics of interaction in the host contact network is also defined by other set of rules. We believe that our approach in the development of this framework can inspire similar approaches to study other types of networks of networks.

In the next section we provide basic insights about the tools chosen to develop the simulation framework, followed by the proper study. In the study, we explain in detail the motivation about the biological system we aim to reproduce, providing information, step by step, about how we adapted the biological model to the computational one.

5.1 Framework for Large-Scale Simulations

The advances in Network Science have allowed us to better understand complex systems, but sometimes we need more than statistical evaluations about dynamics and structure, because sometimes we do not have the knowledge about the real structure and dynamics of those systems. Due to the lack of real data, to achieve that understanding we need to rely on large-scale simulations. We can simulate dynamics by giving rules of interaction between vertexes through the edges, and simulate the structure of the network relying on different topologies.

In this chapter the goal is to provide a simulation framework able to simulate the evolution of bacterial populations over host contact networks. The aim is to observe what impact the structure of the host contact network may have on the population diversity. The simulation process takes into account both evolutionary dynamics in each population and the dynamics over the host contact network, where each population may interact with others accordingly to a given rule. These interactions are represented by the properties of the edges of the host contact network.

To conduct large scale simulations faster and more efficiently, we must optimize the underlying heavy computational processes. To achieve this we can take advantage of High Performance Computing (HPC) systems, parallelizing the simulations whenever possible. Given that we are dealing with highly parallelizable tasks and given the rather simple abstraction provided by the MapReduce paradigm, we adopt it here and we rely on the implementation provided by Apache Spark and GraphX API. Before presenting our study, we provide the basic insights about MapReduce, Apache Spark and GraphX API.

MapReduce [121] is a high-level programming model, proposed to address embarrassingly parallel data processing problems [51], that aims to process and generate large scale data in a parallel, distributed or in a cluster system. In this model, the challenge of parallel programming is addressed by providing an abstraction that isolates the developer from system-level details. There are well-defined interfaces which allow the concepts separation of what computations are to perform and how those computations are actually managed on a cluster of machines. This abstraction comes from the map and reduce primitives in functional programming and allows to parallelize large computations easily. The programmer has only to focus on the code that solves the problem without concerning with parallel code.

With the development of the Hadoop Framework, an open source implementation of the MapReduce programming model originally proposed by Google [51] designed for cheap commodity hardware, but with high fault tolerance, a new light over parallelization problems arised and has been widely used for big data processing and machine learning algorithms. Recently, Apache Spark [197] came with a new approach of MapReduce paradigm, providing a high level API available for several languages, and provides new levels of abstraction exceeding some limitations of Apache Hadoop.

While most MapReduce [121] approaches, as in Hadoop, are built around acyclic data flow model, Apache Spark provides an extended MapReduce model that enables the creation of iterative pro-

grams, maintaining the scalability and fault tolerance of MapReduce, through their Resilient Distributed Datasets [196]. These RDDs are fault-tolerant, parallel data structures that make it possible to persist intermediate results in memory, manage how they are partitioned to optimize data placement, and provide a rich set of operators to apply on the data.

To address graph-parallel problems, Apache Spark provides a proper API, the GraphX API [194]. GraphX allows the user to work with graphs in a transparent manner. With GraphX, graph-parallel and data-parallel computation are possible with a single composable API where both data and graph can be viewed as collections (RDDs) without data duplication. One of the great advantages in using GraphX is that with its PropertyGraph we can attribute properties to the vertexes or edges as a pair of RDDs, having members to access both vertexes and edges, separately.

Details on how we adapted our simulation model to this programming model are in Section 5.2.2.

5.2 Evolution and Diversity of Bacterial Populations

5.2.1 Motivation

Understanding a given bacterial species population structure and how it is shaped by genetic forces of mutation and recombination is vital for interpreting the response of bacterial populations to selection pressures, such as antibiotic treatment or vaccination [152]. In this context, large-scale studies are fundamental not only to understand such processes, but also to validate both models and methods, such as phylogenetic inference algorithms, and to understand how the sampling that is commonly done can affect or bias our perception of the true population structure. However, it is often difficult to execute such large-scale studies with real pathogen samples due to economic and practical reasons. Moreover, most publicly available datasets are biased to their original studies. Hence, given the model complexity and required population sizes, large-scale simulations are the most convenient way to address this issue.

Previous studies have shown that observed population genetic structure of several important human pathogens, such as *Streptococcus pneumoniae* and *Neisseria meningitidis*, can be explained using a simple evolutionary model [77–79, 91]. This model was based on neutral mutational drift and incorporated recombination events, but was tested only for panmictic populations, *i.e.* all the individuals in the model could freely exchange DNA through recombination events. Although this simple evolutionary model works well for local populations, at a “microepidemic” level, its predictions no longer fit observed genetic relationships of large and widely distributed bacterial populations. With the increasing volume of data obtained with sequence based typing methods, such as Multi-Locus Sequence Typing (MLST) [125] (currently the gold standard for epidemiological surveillance for many bacterial species), a much more complex pattern emerges, that cannot be explained solely by the simple “microepidemic” assumption.

The evolution of transmissible bacteria occurs by mutation and recombination, and is influenced

by epidemiological as well as molecular processes. These aspects are fundamental in the process of strain diversification [169], and as a mechanism by which strains acquire virulence factors or resistance determinants [134]. On the other hand, microbial evolution is also influenced by the environment and by host contact networks, which modulates the spread of microbial pathogens. The study of the impact of host contact network topologies, and associated transmission ratios, on bacterial population evolution and genetic diversity becomes then relevant, increasing the model complexity.

Large-scale simulations are, however, computationally demanding, in particular when model complexity increases. In this work, we consider an extension of the above simple evolutionary model by incorporating the underlying host contact network. We propose an extension of the simulation framework for large bacterial populations presented by [175], which implements the Wright-Fisher model [180], on top of Apache Spark [197], making use of both MapReduce programming model and GraphX API [194]. This extended version includes improvements to the code, an implementation of the Simpson's Index of Diversity [163] and a more realistic parametrization, which is used to analyze population diversity under different input parameters.

We center our discussion in two main aspects: how these large-scale simulations benefit from parallelization, evaluating inherent parallelism limits and drawing conclusions on the relation between cluster computing power and simulations speedup; and how bacterial populations are affected by network topology and inherent mutation and recombination rates. We used Google Cloud Platform to conduct our experiments and to analyze the performance of each simulation for each network topology.

5.2.2 Biological and Computational Models

While the field of contact network epidemiology is growing fast, it has become critical to develop models and tools to address above problem. *SparkNetSim* is a parameterizable framework for simulating the evolution of large-scale bacterial populations over complex host contact networks. We focus on bacterial population genetics where isolates are represented as typing profiles, which encodes for a specific genetic lineage. There are several typing methods and, in this work, we consider the Multi-Locus Sequence Typing (MLST) technique. In MLST, DNA sequences are obtained for a set of typically seven housekeeping loci, and different sequences identified at each locus are assigned as different alleles through a unique identifier [125]. Nowadays, it is common to find profiles with hundreds of loci.

In our simulations, each isolate is represented through a profile that may be subject to transformations along time, under the influence of genetic events, namely mutation and recombination, which can be in turn modulated by the environment. Let each strain in a bacterial population be then characterized by a profile, with a profile being defined by the combination of its alleles, a vector of labels. All profiles have the same length and, for each position, different labels among profiles mean different alleles.

In our simulator, the environment is represented by the host contact network. Given such a network,

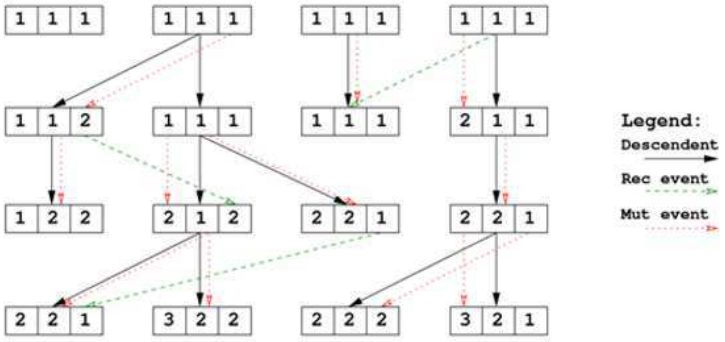


Figure 5.1: Example of IAM. In this example each individual is represented by three alleles. Each allele has a corresponding identifier. When a mutation occur, the allele gains a new unique identifier. When a recombination takes place between two individuals, one allele from one individual is copied for the same position in the other individual.

the bacterial population at each host evolves according to a neutral evolutionary model [78], which, in turn, based on the neutral infinite alleles model (IAM) [106] (see Figure 5.1). It assumes that genetic events do not contribute to the fitness of the individual and, therefore, all individuals in a population have an equal chance of reproducing and being subjected to genetic events. Under IAM, mutation always generates a new allele, leading to new profiles, also known as sequence types (ST). Recombination, on the other hand, introduces an existing allele randomly selected from the isolates present in the previous generation, which may lead to novel allelic profiles, i.e., sequence types, or to the reappearance of existing ones. Mutation and recombination occur independently, with each event being rare and mutation taking precedence over recombination. A new generation is obtained at each step, leading to non-overlapping generations.

The interactions between hosts take place by allowing pathogens to migrate from one host to another. Migration occurs according to the contact network topology, the migration frequency and the edge transmission probabilities are defined by the user. After each migration phase, the population at each host is obtained through sampling with replacement from the set of both the individuals already at the host and those that migrated to it (see Figure 5.2).

We should note that the model based on IAM described above is also known as the neutral Wright-Fisher model [180], where equal fitness means that all individuals can be picked as a parent with the same probability.

Since we are interested in large-scale simulations, we rely on MapReduce programming model on top of Apache Spark [197] and GraphX [194] to parallelize and scale up our simulations. The use of the MapReduce programming model for solving a problem requires redesigning algorithms around *Map* and *Reduce*. On the other hand, since we are exploiting the parallelism among mappers and reducers, running on top of a shared distributed file system, special care is required to avoid synchronization issues. Note that although MapReduce, and in our particular case Apache Spark, hide most of synchronization

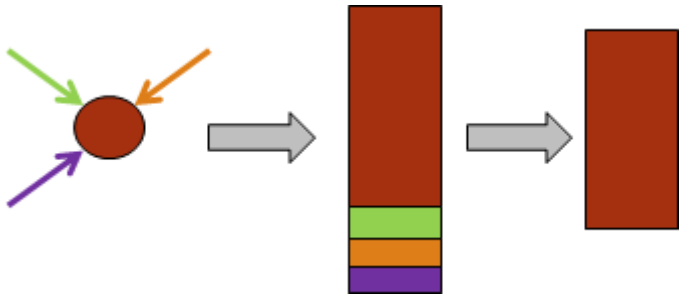


Figure 5.2: Sampling process after exchanges. When exchanges occur, each population joins to its own individuals a given quantity of individuals from its neighbours. Given the evolutionary model, and the constant size of each population, the sampling process consists in creating a pool with all the individuals and choosing, randomly with replacement, the amount of individuals corresponding to the population size.

issues, wrongly design algorithms may not be able to benefit from inherent parallelism.

It is then important to analyze some basic requirements, knowing that in the *Map* phase a function is applied to many key-value pairs in parallel, generating another set of key-value pairs, and that in the *Reduce* phase those intermediate key-value pairs are aggregated by key and a function is applied to each set of values, generating new key-value pairs. Returning to our simulation problem, we have two main tasks: (1) the evolution of each population at each node where mutation and recombination take place; and (2) the exchanges between nodes and the replacement of each population taking into account the samples of the populations migrated from the connected neighbors.

For the first task, although each host has its own population, evolving independently, we must take care of assigning unique global identifiers on mutation events, and sampling STs uniformly among each host population as allele donors on recombination events. We address the first issue by introducing a special schema for generating unique allele identifiers (see implementation for details). The second issue implies grouping each population for each host, a step that we cannot avoid and that will lead to some performance lost.

For the second task, each node has to receive a migrating sample of the current population of its neighbors, mix migrating individual with its own population, and create a new population through sampling. Besides being a simple problem when thinking about the MapReduce model, we can identify some similarities between this process and PageRank [138] or Label Propagation [198] problems. Both problems rely on exchanging information among neighbors to update their state, and both have already been implemented using MapReduce and, in particular, making use of GraphX.

5.2.3 Implementation

Let $G = (V, E)$ be a connected and weighted graph, with $|V|$ nodes and $|E|$ edges, and with an edge transmission probability function $w : E \to \mathbb{R}$. Let t be the number of times that the sequence of number of evolutions followed by the number of exchanges happens.

Algorithm 4: Workflow of the evolutionary model.

```
for each time t do
    for i = 0 to evols do
        evolution of each population
    end
    for j = 0 to exch do
        run exchanges between nodes
    end
end
```

The simulator[1] takes several parameters: (1) the population size of each node p; (2) the file containing the populations; (3) the file containing the network; (4) mutation rate per allele per generation, m; (5) recombination rate per allele per generation, r; (6) number of evolutions, $evols$; (7) number of exchanges, $exchs$; (8) number of times for the cycle $evols$ followed by $exchs$ to happen, t; (9) frequency of sampling/writing on disk s; (10) local or cluster mode; (11) number of partitions; (12) partition to be applied on the Graph; (13) directory to write the results. Each iteration of evolution is considered as one generation. The workflow consists in two main steps: (i) sequence of evolutions for each population, followed by (ii) sequence of exchanges between nodes. This workflow is performed t times (See Algorithn 4.

The first step consists in evolving each population according to the Wright-Fisher model. For each individual, an allele can be subject to both mutation and recombination events with probabilities m and r, respectively. In the second step, individuals migrate among host populations as follows: for each host u, given a neighbor v of u, u samples a given proportion of its population, according to the edge transmission probability $w(u, v)$, and sends it to v; each host receives the population samples sent by its neighbors and creates a pool with those samples mixed with its own population; a new population is built, with the same size as the previous one, but where individuals are chosen randomly from the population pool obtained in the previous step. At the end of this process, all populations are persisted on secondary storage.

Given several samples along generations, we can then evaluate and validate several models and parameters. In this work we focus on diversity changes along time, leading to some exciting results since we are not explicitly considering selection mechanisms. It is possible to calculate the diversity with the Simpson's Index of Diversity (SID) which is the probability that two individuals randomly selected from a sample will belong to the same species [163].

Let us explain how simulations and the computation of SID can be implemented using Apache Spark and GraphX. This will allow us to conduct large-scale simulations and analyses.

For the simulations of bacterial populations we need two input files. The first input file contains the bacterial population for all hosts. This file has an individual per line represented as a *key/value* pair,

[1] The complete implementation discussed here can be found at `https://bitbucket.org/steixeira/sparknetsim/`.

where the *key* corresponds to its host/node identifier and the *value* corresponds to the sequence of its alleles. Although alleles are usually characterized by an integer in real datasets, we need a different characterization as discussed above since we must assign unique global identifiers on mutation events.

Each individual is then characterized by an array of strings, where each string is an allele in the form $X.Y.Z$, where X corresponds to the identifier (ID) of the node/host where the mutation occurs, Y is the generation id, and Z is the number corresponding to the Z-th mutation event of that generation in that node. With this change we guarantee the requirements of the IAM regarding allele uniqueness on mutation events.

The second input file is the network file which contains an edge list with transmission probabilities, i.e., a triple per line with the source u, the destination v and the fraction $w(u, v)$ of the population to be sent from the u to v. Note that, since graphs are assumed to be directed in GraphX, if we want an undirected network, then the edge list must contain edges in both directions. For the exchange process among neighbors, it is necessary for the Map phase that each node sends to itself its own population. Each node must have then an edge to itself with a probability of transmission 1.0. We load the input files with the $SparkContext.textFile$ method, which maps the inputs into RDDs. This method allows the definition of a minimum number of partitions for the data, that is received as a parameter and that must adapted according to our cluster configuration.

Evolutions and exchanges among nodes can be implemented as independent Map and $Reduce$ tasks. Listing 5.1 illustrates the following behavior with some partial Scala . Full code is available in

After loading the input files into an RDDs, we apply the $groupByKey$ transformation in the RDD of the population to guarantee that each population is in the same data structure. This is a requirement to perform the Wright-Fisher model, as recombination demands elements from the same population to be recombined. This transformation, when called on a dataset of (K, V) pairs, returns a dataset of $(K, Iterable\langle V\rangle)$ pairs. Once we have each population gathered in the same data structure, we perform a Map transformation to make each population evolve in parallel. At the end of each iteration we have to cache intermediate results and to apply $collect$ to guarantee that the new RDD contains the new population. We sample then the populations and we persist them using the operation $saveAsTextFile$.

For the exchange process, we build a $PropertyGraph$ using the RDD of the population and the RDD with the edges from the host-contact network. We also allow the user to define which $GraphPartition$ to use as one of the arguments. If the parameter is 0, then no $GraphPartition$ is used; if it is 1, then we use the strategy $PartitionBy2D$. As we discuss later in the experimental evaluation, this seems to be the best partitioning method for scale-free networks (one of the network topologies discussed below).

Once the $PropertyGraph$ has been created, we use the *triplets* view which allows the access to both node and edge properties, i.e., the populations and the fraction of each population to be sent, respectively. At each exchange step, we create the samples from each source node, represented as a

Listing 5.1: Evolutionary process.

```
val popData = sc.textFile(populations,npartitions).cache()
val netData = sc.textFile(network,npartitions).cache()
var individualsRDD = popData.map(*process input file*).groupByKey().cache()

//Evolutions
val nextpop = individualsRDD.map( i =>
        val idpop = i._1
        val population = i._2
        for (each indidivual in population){
            //Recombintation | //Mutation
            population.update(i, individual)}
        (idpop, population)}
nextpop.map(*format output as desired*).saveAsTextFile(outputdir)
individualsRDD = nextpop

//Exchanges
val newpop = graphpopulation.triplets.map( t =>
        //generate a collection with the proportion
        //of population to send
        (dest, fractionofpopulation)
    }.reduceByKey(_ ++ _).map( i =>
        //create the new populations by sampling
        (key, newpopulation)}
newpop.map(*format output as desired*).saveAsTextFile(outputdir)
individualsRDD = newpop
```

new RDD with the key being the destiny node and the value being the sample. Another $reduceByKey$ transformation with the concatenation operator ($++$) allows us to joins all the samples with the same key, providing a collection with all pools of elements from which new populations will be sampled. Each new population is generated through sampling with a Map transformation.At the end of this process, we also call the operation $saveAsTextFile$ over the RDD of the new populations.

For the calculation of the SID, we need an input file where each line is in the format of [$generation\_node\ individual$] (this is also the output format of our simulator). After loading the data into an RDD and also defining in how many partitions we want to partition our data, we apply the $groupByKey$ transformation to aggregate values for each node, per generation. We use then the Map transformation to process each node per generation in parallel. It is now straightforward to compute the SID using operators $groupBy(identity)$, which groups each unique individual, and then $mapValues$ to count how many unique sequences there are. The remain calculus is the SIDs formula (see Listing 5.2).

We should note that when we run large-scale simulations, the output can easily reach dozens of gigabytes. And the SID calculation with Apache Spark for an output size of 30GB takes seconds. In the end, we use again $saveAsTextFile$ to write on disk the SID of each node, for each generation.

Listing 5.2: Simpson's Index Diversity Calculation.

```
var individuals = popData.map{ s =>
    val indiv = s.split("\t")
    (indiv(0), indiv(1))
}.groupByKey().map{ geracao =>
    val indiv = geracao._2.toVector
    val ocorrencias = indiv.groupBy(identity).mapValues(_.size)
    val SIDn = ocorrencias.map(i => (i._2 * (i._2 - 1))).reduce(_+_)
    val SID : List[String] = List(geracao._1 + "\t" + (1 - (SIDn / popSIDTotal)))
    (geracao._1, SID)
}
individuals.map(f => f._2.mkString(" ")).saveAsTextFile(outputdir + "outputSID")
```

5.2.4 Experimental Analysis

In this section we present experimental results concerning both performance evaluation and population diversity growth analysis. These results rely on homogeneous and heterogeneous host-contact networks. For homogeneous networks we used fully connected networks, representing well-mixed populations, and regular networks. In the latter, we consider regular ring lattices, where each node has the same number of neighbours and is connected to those that are closest [190]. For heterogeneous we consider scale-free networks generated by the B-A model [13] and the DMS Model [56], the second one portraying higher clustering coefficient (for more details see Chapter 2). Except for the fully connected network (where all nodes have degree $|V| - 1$), all networks have average degree 4.

Performance Evaluation

Apache Spark can run either on local or on cluster modes. We relied on both modes for our experiments to compare how running time scales, parameterizing simulations with different population sizes and different graphs. Experiments were conducted in a cluster hosted at Google Cloud Platform[2], configured with different number of workers: 2, 4, 8 and 16. Each worker is an Intel(R) Xeon(R) CPU @ 2.50GHz with 4 cores and 16GB of RAM. To achieve the best results possible, in each experiment we partitioned the input in as many partitions as the number of the cores available. The local setup was performed in a worker node, using only one core.

Given the evolutionary model coupled with a host contact network, leading to both evolution and interaction between populations, the operations in local mode take considerable time, with each step being executed sequentially. Also, it scales reasonably well when in cluster mode. For a precise comparison, in what concerns scalability, the simulator parameters are fixed as follows: the size of the population per node is 1000, the mutation rate is 0.001, the recombination rate is 0.01, the number of evolutions per time

[2]https://cloud.google.com/

		Mode				
Topology	**Size (n)**	**Local 1 core**	**2 workers 8 cores**	**4 workers 16 cores**	**8 workers 32 cores**	**16 workers 64 cores**
Clique	200	2466.4	225.0	218.2	93.8	89.4
	500	7315.4	852.6	446.2	228.4	152.8
	1000	16994.6	3827.8	1077.0	515.4	315.0
	2000	39427.5	13436.2	4906.6	1507.4	746.2
Regular	200	1487.8	149.2	157.4	73.8	69.0
	500	2271.2	437.2	385.2	137.6	97.4
	1000	4134.4	1576.6	862.6	258.6	163.6
	2000	6912.4	5037.6	1746.8	534.6	303.4
B-A	200	1456.0	153.4	145.4	75.4	69.0
	500	2575.2	471.8	383.2	134.0	102.2
	1000	4342.2	1602.8	867.2	253.8	164.6
	2000	6564.2	5038.4	1801.4	549.0	305.2
DMS	200	1448.2	150.0	142.6	75.6	67.2
	500	2829.4	442.2	391.8	135.8	92.6
	1000	4377.8	1600.0	850.6	248.0	163.8
	2000	6682.2	4692.0	1825.6	547.0	284.0

Table 5.1: Running time in seconds for different topologies and network sizes.

step is 25, the number of exchanges per time step is 1, the number of time steps is 10, the frequency for writing on disk is 10 generations, and the transmission probability is 0.01 for all edges.

The first observation is that writing on disk is the most costly operation. This is even more noticeable in the new version of the simulator presented in this paper, where data is persisted for each main iteration (and used afterwards to compute for instance the SID). We address this issue by using the $saveAsTextFile$ method available in Apache Spark, exploiting the ability to parallelize write operations on the underlying distributed file system (HDFS) through data partitioning.

We consider three different network topologies as described before: cliques, regular networks and two scale-free networks [13, 56] sharing the same degree distribution, yet portraying different clustering coefficients. The clique topology leads to the highest computational cost when performing exchanges. Scale-free networks are more realistic for host contact networks, but being very sparse lead to much less work during exchanges. Tests were run for fully built topologies, an average degree of 4, and different network sizes. The results averaged over 10 runs are presented in Table 5.1 and in Figure 5.3.

We can observe that the speedup becomes more evident as networks grow in size. Running in cluster mode, and increasing the number of workers and cores, results in a significantly reduction in the running time, namely for larger networks. The fact that we can compute the population evolution at each node independently seems to be exploited as expected. The same happens with the exchange process among nodes populations.

When designing parallel algorithms, one of the analyses that should be done is to estimate the relation between achievable speedups and the number of workers/cores. Amdahl's law is useful in

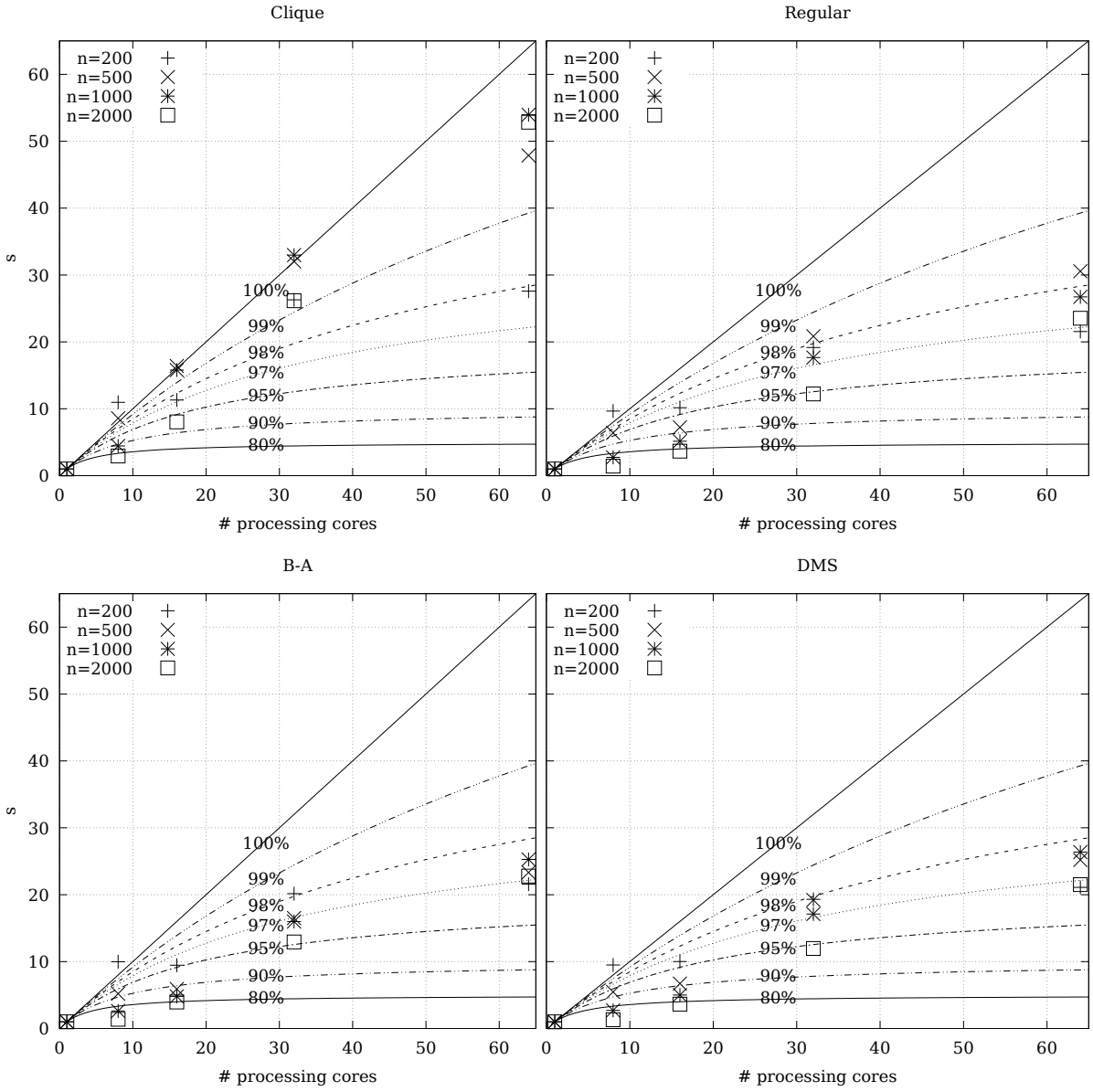
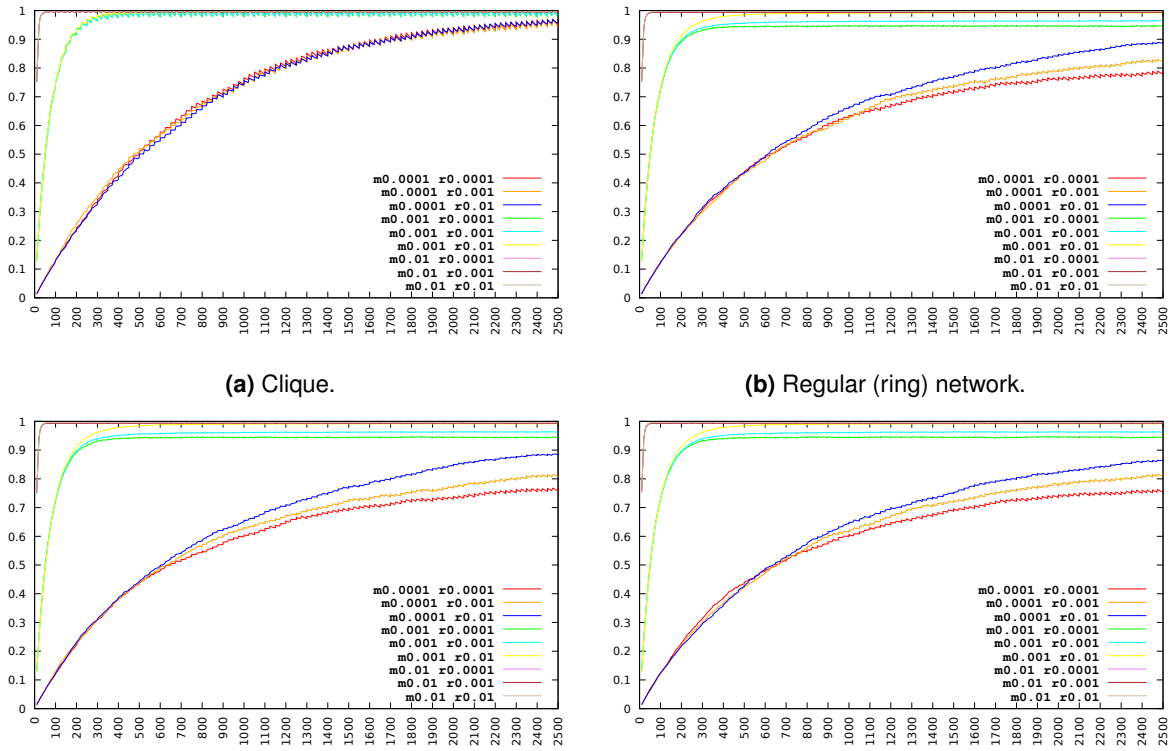


Figure 5.3: Speedup as a function of the number of available cores for cliques, regular networks, B-A and DMS scale-free networks, with different network sizes (n). The curves are provided by Amdahl's law [7], where the percentage corresponds to the fraction that is infinitely parallelizable.

this context to both understand results and project expected speedups [7]. The main points are that we should only optimize if the fraction that can be optimized constitutes a large portion of the overall time, and that if the optimization is effective, the obtained speedup is largely determined by the strictly sequential fraction. Although this fraction constituted only a small fraction of the initial time, since it cannot be optimized, it will represent a larger fraction as more parallelism is allowed. Given k workers (cores) and a program that spends a fraction f of time on operations that are infinitely parallelizable, and the remaining fraction $1 - f$ on strictly sequential operations, the overall speedup is given by $1/((1 - f) + f/m)$. In Figure 5.3 we can observe that, as we increase the size of the clique networks, we obtain a higher speedup as we increase the number of workers. According to Amdahl's law, we are observing $f > 99\%$. For the scale-free networks, because they are much sparser than cliques, with less work performed on exchange, we observe about $f = 97\%$ and speedups are small above 32 cores. This seems to point out that the exchange process is highly parallelizable, independently of the network size, which is an important observation if simulations with much larger networks are desirable. Note also that the running time grows almost linearly as we increase the number of nodes (see Table 5.1). Although expected, this observation shows that the evolution of populations occurs independently in our implementation, where special care was taken in what concerns the parallelization of recombination and mutation events, while using the IAM model.

Population Diversity Analysis

We now analyze the impact of the host contact network topology on the bacterial population genetic diversity for different rates of mutation and recombination. We are interested to see if strain persistence and local evolutionary events, reflecting local expansion, and limitation of genetic exchange due to the host-contact network can lead to observable variations of SID values that can erroneously be attributed to selection events. For that, we run simulations for 2500 generations (with cycles of 25 evolutions followed by one exchange between nodes), and compute the Simpson's Index of Diversity (SID) for each node. We used four network topologies of size 1000, as described before: a clique, a regular (ring) network, and two scale-free networks, following the B-A model and the DMS model, with power-law degree distributions (see Chapter 2). All networks (besides the clique), share an average degree of 4. The population size was 1000 per node, with each simulation containing 1000000 individuals. Over all, the mutation and recombination rate took values of 0.0001, 0.001 and 0.01. Figures 5.4 and 5.5 depict the results. In Figure 5.4, each image contains – for the four different topologies and nine different combinations of mutation and recombination rates – the average SID per generation. In Figure 5.5 each image contains – for the four different topologies and nine different combinations of mutation and recombination rates – the minimum, 1^{st} quantile, 2^{nd} quantile (median), 3^{rd} quantile and maximum SID per generation, presented as a box-and-whisker plot.



(a) Clique.

(b) Regular (ring) network.

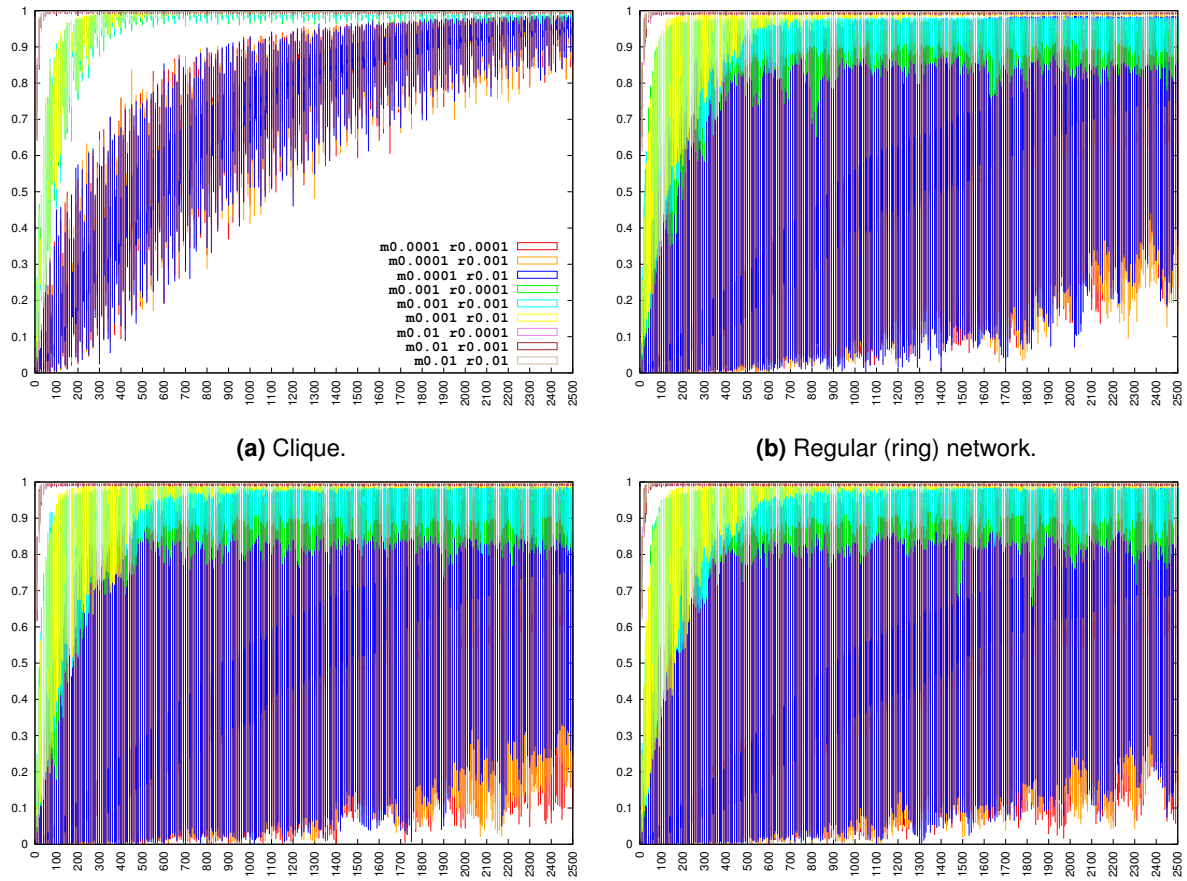
(c) B-A scale-free network.

(d) DMS scale-free network.

Figure 5.4: Population diversity analysis. Each subfigure (representing a different topology) contains a plot representing the average of the SID per generation, for each combination of Mutation and Recombination rates. See main text for details.

We first analyze the population diversity evaluation in a fully connected network (Figures 5.4a and 5.5a), using it as a baseline, as it represents a well-mixed population. It is possible to observe that with higher rates of mutation, the rates of recombination and the effect of exchanges between nodes have no significant effect and the SID proliferates up to the point it stabilizes at the maximum value. For lower probabilities of mutation, there is a slight difference in what concerns the effect of the recombination rate and the effect of exchanges between nodes. When the recombination rate is also small, the SID increases slowly and, when it reaches the maximum value, small and recurrent fluctuations occur over time due to the exchange process.

As we increase the degree heterogeneity, we begin to see some differences and the exchanges process have a significant impact. The only constant observation for all topologies is that when the mutation rate is high, the diversity grows faster, reaching a stable state. When we adopt smaller values for mutation and recombination rates, the SID show pronounced fluctuations each time an exchange is executed. The most interesting case is when mutation and recombination rates are both 0.001, as we can see in Figures 5.4b and 5.5b, 5.4c and 5.5c, and 5.4d and 5.5d. In these cases, it is possible to observe that the interval of values that SID takes reflects that some nodes lose their diversity through time, even



(a) Clique.

(b) Regular (ring) network.

(c) B-A scale-free network.

(d) DMS scale-free network.

Figure 5.5: Population diversity analysis. Each subfigure (representing a different topology) contains a box-and-whisker plot (depicting the minimum, 1^{st} quantile, 2^{nd} quantile (median), 3^{rd} quantile and maximum values) of the SID per generation, for each combination of Mutation and Recombination rates. See main text for details.

without any selection mechanism, as it is the case here. This can erroneously be attributed to selection events, suggesting that distinction between drift and selection is essential if we aim to understand natural evolutionary processes.

5.3 Discussion

In this chapter we propose a framework for large-scale simulation of the evolution of bacterial population over complex networks, which can be run locally or in a distributed environment. We rely on the Wright-Fisher evolutionary model, and on the Map Reduce computational model on top of Apache Spark and GraphX API. The results show that we can run simulations with 1000000 individuals, for 250 generations in less than three minutes, in a distributed computing environment. Our results also point out that,

for scale-free host-contact networks – which represent most real-world scenarios – and according to Amdahl's law, using more than 32 processing cores does not lead to significant improvements for realistic simulations. Since the evolution of populations can occur independently, we may observe further gains from a large number of processing cores for larger populations, exploring the inherent parallelism. We leave this analysis for future work.

Platforms like the one proposed here have a wide range of applications. Large-scale studies and simulations are fundamental not only to understand bacterial population structure and the response of bacterial populations to selection pressures, but also to validate both models and methods, such as phylogenetic inference algorithms, and to understand how the sampling that is commonly done can affect or bias our perception of the true population structure. As an illustration of such applicability, we address the long-standing problem of genetic diversity due to neutral drift. According to our results, fluctuations in population diversity can be explained by neutral drift only, without selection pressure, strongly depending both on the average degree and degree heterogeneity of the host-contact network. As far as we known, such phenomena have not been observed before.

6

Final Remarks

The universe of complex networks is a fascinating one. The amount of possibilities to explore the discovery of structure and dynamics of different complex systems is almost intangible. Nonetheless, each contribution is a step forward to better understand the systems around us which have so much impact in our daily life. In this thesis we presented different types of approaches to analyze complex networks in which edges play a fundamental role. Despite of all the achievements with node based approaches, we believe, and we show in this thesis, that an edge perspective has yet much to provide.

Our contributions started with the proposal of a new edge centrality measure, Spanning Edge Betweenness. In Chapter 3 we provided both theory and methods (Sections 3.1 and 3.2) that allow us to use this edge property to analyze confidence in phylogenetic trees (Section 3.3) and network robustness (Section 3.4). As shown in the two studies, this edge-based approach can provide relevant information about the structure of complex networks.

Working with spanning trees and creating the Spanning Edge Betweenness measure raised some questions with respect to the importance of the structure of a network in problems which can be study with an edge percolation process, but also raised the issue of generating uniformly, as efficiently as possible, a random spanning tree. Experimental results were achieved in [153].

Reinforcing our interest in studying networks from an edge perspective, we plan to explore spanning edge betweenness in the context of network resilience and the maintenance of network global connectivity, following the line of research of Qian *et al.* [195] and comparing spanning edge betweenness with other edge centrality measures. Since spanning edge betweenness gives direct information about the importance of a link, on further research we plan to investigate other application fields as epidemic spreading, identifying which links are critical in the spreading process, following some of the ideas introduced in [88]. Reaching this point, for future work, we also identify the need for a proper survey on edge centrality measures, their applications, and which are best for each context.

In Chapter 4 we looked to networks at a mesoscale observing that capturing local patterns can explain some system's dynamics. The local patterns analyzed are centered in structural social balance theory (Section 4.1). This theory was initially developed to explain how friendship and enmity shaped balanced relations. It has been fascinating to observe that there are other systems that can be modeled as signed networks, taking advantage of the same framework. We started by proposing a model that prove that social balance, observed in real data, can emerge from simple mechanisms of peer influence (Section 4.2). Next, in Section 4.3, we studied how social dilemmas shape social balance. We presented three of the most known social dilemmas and also the models and methods for solving them with social and individual learning. Then, we presented a model to build the corresponding signed network and we concluded that social balance can strongly emerge from cooperation through social learning.

Finally, given the polarity natureof structural balance, we presented a third study of structural balance applied to financial networks. Our interest was to observe if the frequency of structural balance

patterns could mimic one of the most used fear measure in economics, the VIX Index. In Section 4.4 we revealed that there is a strong correlation between balanced motifs and the fear index, highlighting that the frequency of motifs analysis is simpler to conduct than the calculus of VIX Index.

In what concerns the social aspect of structural balance, given the conclusions of Section 4.2 and Section 4.3, in the future it is of most interest to study the interplay between peer influence, cooperation and social balance. Recently He *et al.* [97] studied the evolution of cooperation under the impact of structural balance. Following some of the ideas of this work, we are interested in exploring the effect of adding the peer influence model presented to the cooperation/defection decision process of the agent. Instead of taking into account only the the payoff matrix on the evolutionary model, we aim to add a new parameter to analyze the emotional influence on the strategies chosen.

For the financial networks, further studies must be pursed for a better understanding of the financial world from a network based perspective. It will be relevant to assess whose firms are more often in the motifs that accurately replicate the VIX index and also to acknowledge the impact of such ties throughout time. Indeed, the stability and time-evolution of this class of ties remain, to a large extent, an open question. Each of these sub-graphs, defined by a particular pattern of interactions between vertices, may reflect a meso-scale pattern that will latter lead to particular global financial patterns. In particular, it is relevant to identify or engineer sub-graphs that are more resilient and stable than the rest of the financial network, creating a portfolio with a lower likelihood of suffering in times of crisis or crashes. As future work, it seems relevant to understand if those measures are still reliable at anticipating and predicting future swings in market instability and what are the main contributors of financial volatility and their relation with size, sector, cumulative performance, among others, calling for novel approaches and combinations of network science and other predictive tools. For instance, it would be interesting to study how network measures can complement other data mining tools. Furthermore, producing similar studies in other markets, regions of the globe or joining them together might be enlightening at detailing the responsible for global movements or those that, despite its characteristics, may easily spread throughout the financial market network.

In Chapter 5 we presented a large-scale simulation framework for the evolution of bacterial populations over host contact network. Given the evolutionary Wright-Fisher model and a host contact network, we were able to adapt the evolutionary model to a parallel programming model. We used the MapReduce programming model on top of Apache Spark and its GraphX API, that allowed us to easily access the edge properties that shaped the interactions between populations. We were able to run the simulations efficiently and in a distributed environment, taking advantage of powerful resources as Google Cloud Platform.

This framework has the potential to be used in a wide range of applications. Large-scale studies and simulations are fundamental not only to understand bacterial population structure and the response

of bacterial populations to selection pressures, but also to validate both models and methods, such as phylogenetic inference algorithms, and to understand how the sampling that is commonly done can affect or bias our perception of the true population structure. Moreover, we believe that this framework is a first step to develop a model able to simulate any type of networks of networks, revealing more about the dynamics complex systems in the most diverse domains. As future work we plan to investigate how can we abstract the computational model to allow heterogeneity at the node level in the contact network. Our aim is to extend the presented framework to one where we can abstract the *evolution of each population* step in Algorithm 4 to a function that depends on what a vertex represents, e.g. a network or an agent. Once again we will be confronted with the challenge of how to further parallelize such function.

This thesis provided many research opportunities and a new study is already in motion. Extending our interest in the study of observing local patterns ant their impact on human behaviour, we plan to study how fairness emerges in societies through the Ultimatum Game (UG) played in groups, instead of peer to peer [160, 179]. Fairness has a profound impact on human decisions and individuals often prefer fair – over payoff maximizing – outcomes [72]. This evidence was pointed several times, often resorting to behavioural experiments with the UG: a Proposer decides how to divide a given resource with a Responder and the game only yields payoff to the participants if the Responder accepts the proposal. Strangely, Proposers tend to sacrifice their share by offering high proposals and Responders often prefer to earn nothing rather than accepting unfair divisions [72]. These counter-intuitive results motivated several theoretical models that aimed at justifying, mathematically, the evolution of fair intentions in human behaviour. In most cases, however, the roles in UG are assumed to be symmetric: each node has an even probability of being the Proposer or the Responder. Also often, both roles are played simultaneously. These assumptions are naturally at odds with reality, where being the Proposer or Responder depends on characteristics of individuals. Proposers – such as employers, auction first-movers or investors – are in the privileged position of deciding which divisions to offer. The benefits of Proposers are even increased in multiplayer ultimatum games [160]. In that case, Responders often need to divide the offers, thus increasing the gap between Proposers' and Responders' potential earnings.

This leads us to two main questions: *i)* Which criteria should be used to select Proposers within a group? and *ii)* What is the impact of different criteria on the emerging levels of fairness? We plan to analyze Multiplayer Ultimatum Games (MUG) [161] in heterogeneous complex networks, which allows us to test several network properties as base criteria for defining how to select Proposers in a group. Our preliminary results suggest that offering the first move to low-degree nodes balances the natural power of highly connected nodes in scale-free networks, leading to a significant increase in the global levels of fairness. We intend to explore other network measures, as structural power [160]. Also, we want to explore again the idea of adding an emotional layer into the MUG. What would be the effect of playing the MUG in a signed network? Can the influence of our emotions overcome the rational choice at the

proposal acceptance or rejection?

With Network Science we are able to model and study the most diverse complex systems. Nonetheless, we believe that there is room for even more advances if we combine techniques from different fields. As future work, we aim to join complex networks models and tools with machine learning and data mining techniques.

In the field of medicine, we are interesting in pursuing predictive solutions for Neurodegenerative diseases. One study that is about to start is based on the work of Carreiro *et al.* [34]. We will start with Amyotrophic Lateral Sclerosis disease, which is a devastating neurodegenerative disease characterized by a fast progression of muscular denervation atrophy, leading to death in just a few years. We aim to model patients data into networks, exploring possible network measures/patterns that could improve predictive power. One of the main challenges is how to model temporal data and also how to define edges in the network. While nodes represent patients, edges are usually modeled as a distance metric between them. Given that each patient has both static and temporal data, we aim to improve Carreiro *et al.* [34] work, by trying to discretize temporal data, creating static patterns, and to explore new distance metrics, while also contributing to the ongoing challenge of analyzing complex temporal networks [99].

As we have shown, complex systems can be explored under different perspectives. This diversity enriches Network Science and we hope that our contributions find their way to remarkable knowledge that can improve our understanding about the systems that surround us, but also to our quality of life.

Publications and Communications

Publications

1. Teixeira, A. S., Monteiro, P. T., Carriço, J. A., Ramirez, M., & Francisco, A. P. (2013). Spanning edge betweenness. In *Workshop on mining and learning with graphs* (Vol. 24, pp. 27-31). (http://snap.stanford.edu/mlg2013/submissions/mlg2013_submission_14.pdf)

2. Teixeira, A. S., Monteiro, P. T., Carriço, J. A., Ramirez, M., & Francisco, A. P. (2015). Not seeing the forest for the trees: size of the minimum spanning trees (MSTs) forest and branch significance in MST-based phylogenetic analysis. *PloS one*, 10(3), e0119315. (https://doi.org/10.1371/journal.pone.0119315).

3. Teixeira, A. S., Santos, F. C., & Francisco, A. P. (2016). Spanning Edge Betweenness in Practice. In *Complex Networks VII* (pp. 3-10). Springer, Cham. (https://doi.org/10.1007/978-3-319-30569-1_1)

4. Teixeira, A. S., Santos, F. C., & Francisco, A. P. (2017, March). Emergence of Social Balance in Signed Networks. In *Workshop on Complex Networks CompleNet* (pp. 185-192). Springer, Cham. (https://doi.org/10.1007/978-3-319-54241-6_16)

5. Teixeira, A. S., Monteiro, P. T., Carriço, J. A., Santos, F. C., & Francisco, A. P. (2017, August). Using Spark and GraphX to Parallelize Large-Scale Simulations of Bacterial Populations over Host Contact Networks. In *International Conference on Algorithms and Architectures for Parallel Processing* (pp. 591-600). Springer, Cham. (https://doi.org/10.1007/978-3-319-65482-9_44)

6. Teixeira, A. S., Monteiro, P. T., Carriço, J. A., Santos, F. C., & Francisco, A. P. (2018). Large-scale simulations of bacterial populations over complex networks. *Journal of Computational Biology*, 25(8), 850-861. (https://doi.org/10.1089/cmb.2018.0083)

7. Russo, L., Teixeira, A. S., & Francisco, A. P. (2018). Linking and Cutting Spanning Trees. *Algorithms*, 11(4), 53. (http:/https://doi.org/10.3390/a11040053)

8. Souto, P. C., Teixeira, A. S., Francisco, A. P., & Santos, F. C. (2018, December). Capturing Financial Volatility Through Simple Network Measures. In *International Workshop on Complex Networks and their Applications* (pp. 534-546). Springer, Cham. (https://doi.org/10.1007/978-3-030-05414-4_43)

9. Teixeira, A. S., Santos, F. C., Francisco, A. P., & Santos, F. P. (2018, December). Fairness in multiplayer ultimatum games through degree-based role assignment. In Proc. *International Conference on Complex Networks and their Applications* 2018 (extended abstract).

10. Teixeira, A. S., Fernandes, F., & Francisco, A. P. (2018). SpliceTAPyR -— An Efficient Method for Transcriptome Alignment. *International Journal of Foundations of Computer Science*, 29(08), 1297-1310. (https://doi.org/10.1142/S0129054118430049)

Communications

Oral Presentation

1. Emergence of Social Balance in Signed Networks – 8th Workshop on Complex Networks CompleNet 2017 (Dubrovnik, Croatia).

2. Using Spark and GraphX to Parallelize Large-Scale Simulations of Bacterial Populations over Host Contact Networks – 5th InternationalWorkshop on Parallelism in Bioinformatics in the 17th International Conference on Algorithms and Architectures for Parallel Processing 2017 (Helsinki, Finland).

3. Large-scale population studies and link significance in phylogenetic analysis – Computational Biology and Bioinformatics Seminar @ iMM 2018 (Lisbon, Portugal).

4. Link Significance in Phylogenetic Analysis – DSB 2018 – 4th Workshop on Data Structures in Bioinformatics (Helsinki, Finland).

5. Fairness in multiplayer ultimatum games through degree-based role assignment – The 7th International Conference on Complex Networks and Their Applications 2018 (Cambridge, England).

Poster Presentation

1. Spanning edge betweenness in practice – 7th Workshop on Complex Networks CompleNet 2016 (Dijon, France).

2. Large scale simulation of bacterial population evolution over host contact networks – Summer Solstice 2016: 8th International Conference on Discrete Models of Complex Systems (Aveiro, Portugal).

3. Capturing Financial Volatility Through Simple Network Measures – The 7th International Conference on Complex Networks and Their Applications 2018 (Cambridge, England).

Bibliography

[1] AFONSO, G., KOVNER, A., AND SCHOAR, A. Trading partners in the interbank lending market. *Federal Reserve Bank of New York Staff Reports*, 620 (2013).

[2] ALBERT, R., JEONG, H., AND BARABÁSI, A.-L. Internet: Diameter of the world-wide web. *Nature 401*, 6749 (Sept. 1999), 130–131.

[3] ALLEN, F., AND BABUS, A. Networks in finance. *The Network Challenge* (2009), 367–382.

[4] ALLEN, F., BABUS, A., AND CARLETTI, E. Asset commonality, debt maturity and systemic risk. *Journal of Financial Economics 104*, 3 (2012), 519–534.

[5] ALON, N., YUSTER, R., AND ZWICK, U. Finding and counting given length cycles. *Algorithmica 17*, 3 (1997), 209–223.

[6] AMARAL, L. A. N., SCALA, A., BARTHELEMY, M., AND STANLEY, H. E. Classes of small-world networks. *Proceedings of the National Academy of Sciences USA 97*, 21 (2000), 11149–11152.

[7] AMDAHL, G. M. Validity of the single processor approach to achieving large scale computing capabilities. In *Proceedings of the April 18-20, 1967, Spring Joint Computer Conference* (1967), AFIPS '67 (Spring), ACM, pp. 483–485.

[8] ANTAL, T., KRAPIVSKY, P. L., AND REDNER, S. Dynamics of social balance on networks. *Phys. Rev. E 72* (Sep 2005), 036121.

[9] AREF, S., MASON, A. J., AND WILSON, M. C. Computing the line index of balance using integer programming optimisation. In *Optimization Problems in Graph Theory*. Springer, 2018, pp. 65–84.

[10] AREF, S., AND WILSON, M. C. Balance and frustration in signed networks. *Journal of Complex Networks* (2018), cny015.

[11] ASHTIANI, M., SALEHZADEH-YAZDI, A., RAZAGHI-MOGHADAM, Z., HENNIG, H., WOLKENHAUER, O., MIRZAIE, M., AND JAFARI, M. A systematic survey of centrality measures for protein-protein interaction networks. *BMC Systems Biology 12*, 1 (2018), 80.

[12] AXELROD, R., AND HAMILTON, W. D. The evolution of cooperation. *Science 211*, 4489 (1981), 1390–1396.

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

[14] BARABÁSI, A.-L. The origin of bursts and heavy tails in human dynamics. *Nature 435* (2005), 207.

[15] BARABÁSI, A.-L. *Network science*. Cambridge University Press, Cambridge, 2016.

[16] BARDOSCIA, M., BATTISTON, S., CACCIOLI, F., AND CALDARELLI, G. Debtrank: A microscopic foundation for shock propagation. *PloS ONE 10*, 6 (2015), e0130406.

[17] BARDOSCIA, M., BATTISTON, S., CACCIOLI, F., AND CALDARELLI, G. Pathways towards instability in financial networks. *Nature Communications 8* (2017), 14416.

[18] BARDOSCIA, M., CACCIOLI, F., PEROTTI, J. I., VIVALDO, G., AND CALDARELLI, G. Distress propagation in complex networks: the case of non-linear debtrank. *PloS ONE 11*, 10 (2016), e0163825.

[19] BARRAT, A., BARTHELEMY, M., AND VESPIGNANI, A. *Dynamical processes on complex networks*. Cambridge university press, 2008.

[20] BARRETT, S. Self-enforcing international environmental agreements. *Oxford Economic Papers* (1994), 878–894.

[21] BARUCCA, P., BARDOSCIA, M., CACCIOLI, F., D'ERRICO, M., VISENTIN, G., BATTISTON, S., AND CALDARELLI, G. Network valuation in financial systems. *arXiv e-prints* (2016).

[22] BATTISTON, S., CALDARELLI, G., D'ERRICO, M., AND GURCIULLO, S. Leveraging the network: a stress-test framework based on debtrank. *Statistics & Risk Modeling 33*, 3-4 (2016), 117–138.

[23] BATTISTON, S., CALDARELLI, G., MAY, R. M., ROUKNY, T., AND STIGLITZ, J. E. The price of complexity in financial networks. *Proceedings of the National Academy of Sciences USA 113*, 36 (2016), 10031–10036.

[24] BATTISTON, S., PULIGA, M., KAUSHIK, R., TASCA, P., AND CALDARELLI, G. Debtrank: Too central to fail? financial networks, the fed and systemic risk. *Scientific Reports 2* (2012), 541.

[25] BISWAS, A., AND BISWAS, B. Community-based link prediction. *Multimedia Tools and Applications 76*, 18 (Sep 2017), 18619–18639.

[26] BOCCALETTI, S., BIANCONI, G., CRIADO, R., DEL GENIO, C. I., GÓMEZ-GARDENES, J., ROMANCE, M., SENDINA-NADAL, I., WANG, Z., AND ZANIN, M. The structure and dynamics of multilayer networks. *Physics Reports 544*, 1 (2014), 1–122.

[27] BOCCALETTI, S., LATORA, V., MORENO, Y., CHAVEZ, M., AND HWANG, D.-U. Complex networks: Structure and dynamics. *Physics Reports 424*, 4-5 (2006), 175–308.

[28] BODIE, Z., KANE, A., AND MARCUS, A. J. *Investment and portfolio management*. McGraw-Hill Irwin, 2011.

[29] BOGUNÁ, M., PASTOR-SATORRAS, R., AND VESPIGNANI, A. Cut-offs and finite size effects in scale-free networks. *The European Physical Journal B 38*, 2 (2004), 205–209.

[30] BORGATTI, S. P., AND EVERETT., M. G. A graph-theoretic perspective on centrality. *Social Networks 28*, 4 (October 2006), 466–484.

[31] BOSS, M., ELSINGER, H., SUMMER, M., AND THURNER 4, S. Network topology of the interbank market. *Quantitative Finance 4*, 6 (2004), 677–684.

[32] BRODER, A. Z., AND MAYR, E. W. Counting minimum weight spanning trees. *Journal of Algorithms 24*, 1 (1997), 171–176.

[33] BULDYREV, S. V., PARSHANI, R., PAUL, G., STANLEY, H. E., AND HAVLIN, S. Catastrophic cascade of failures in interdependent networks. *Nature 464*, 7291 (2010), 1025.

[34] CARREIRO, A. V., MADEIRA, S. C., AND FRANCISCO, A. P. Unravelling communities of als patients using network mining. In *ACM SIGKDD Workshop on Data Mining in Healtcare* (2013), Citeseer.

[35] CARTWRIGHT, D., AND HARARY, F. Structural balance: a generalization of Heider's theory, 1956.

[36] CASTELLANO, C., FORTUNATO, S., AND LORETO, V. Statistical physics of social dynamics. *Rev. Mod. Phys. 81*, 2 (2009), 591–646.

[37] CHRISTAKIS, N. A., AND FOWLER, J. H. The spread of obesity in a large social network over 32 years. *New England Journal of Medicine 357*, 4 (2007), 370–379.

[38] CHRISTAKIS, N. A., AND FOWLER, J. H. The collective dynamics of smoking in a large social network. *New England Journal of Medicine 358*, 21 (2008), 2249–2258.

[39] CHUNG, F. Graph theory in the information age. *Notices of the AMS 57*, 6 (2010), 726–732.

[40] CORMEN, T. H., LEISERSON, C. E., RIVEST, R. L., AND STEIN, C. *Introduction To Algorithms*. MIT Press, 2001.

[41] COSTA, L. F., RODRIGUES, F. A., TRAVIESO, G., AND VILLAS BOAS, P. R. Characterization of complex networks: A survey of measurements. *Advances in Physics 56*, 1 (2007), 167–242.

[42] COSTENBADER, E., AND VALENTE, T. W. The stability of centrality measures when networks are sampled. *Social networks 25*, 4 (2003), 283–307.

[43] DANZIGER, M. M., BASHAN, A., BEREZIN, Y., SHEKHTMAN, L. M., AND HAVLIN, S. An introduction to interdependent networks. In *International Conference on Nonlinear Dynamics of Electronic Systems* (2014), Springer, pp. 189–202.

[44] DARWIN, C. *On the origin of species by means of natural selection, or the preservation of favoured races in the struggle for life*. John Murray, Albemarle Street, London, 1861.

[45] DAS, K., SAMANTA, S., AND PAL, M. Study on centrality measures in social networks: a survey. *Social Network Analysis and Mining 8*, 1 (2018), 13.

[46] DAVIS, J. A. Clustering and structural balance in graphs. *Human Relations 20*, 2 (1967), 181–187.

[47] DAWES, R. M. Social dilemmas. *Annual Review of Psychology 31*, 1 (1980), 169–193.

[48] DE DOMENICO, M., SOLÉ-RIBALTA, A., COZZO, E., KIVELÄ, M., MORENO, Y., PORTER, M. A., GÓMEZ, S., AND ARENAS, A. Mathematical formulation of multilayer networks. *Physical Review X 3*, 4 (2013), 041022.

[49] DE MEO, P., FERRARA, E., FIUMARA, G., AND RICCIARDELLO, A. A novel measure of edge centrality in social networks. *Know.-Based Syst. 30* (June 2012), 136–150.

[50] DE WAAL, F. *Primates and philosophers: How morality evolved*. Princeton University Press, 2009.

[51] DEAN, J., AND GHEMAWAT, S. Mapreduce: Simplified data processing on large clusters. *Commun. ACM 51*, 1 (2008), 107–113.

[52] DI MURO, M., VALDEZ, L., RÊGO, H. A., BULDYREV, S., STANLEY, H., AND BRAUNSTEIN, L. Cascading failures in interdependent networks with multiple supply-demand links and functionality thresholds. *Scientific Reports 7*, 1 (2017), 15059.

[53] DIESTEL, R. *Graph Theory*. Electronic library of mathematics. Springer, 2006.

[54] DOROGOVTSEV, S. N. *Lectures on complex networks*, vol. 24. Oxford University Press Oxford, 2010.

[55] DOROGOVTSEV, S. N., AND MENDES, J. F. *Evolution of networks: From biological nets to the Internet and WWW*. OUP Oxford, 2013.

[56] DOROGOVTSEV, S. N., MENDES, J. F., AND SAMUKHIN, A. N. Size-dependent degree distribution of a scale-free growing network. *Physical Review E 63*, 6 (2001), 062101.

[57] DOYLE, P. G., AND SNELL, J. L. *Random walks and electric networks*, vol. 22. Mathematical Association of America Washington, DC, 1984.

[58] DUGATKIN, L. A. *The altruism equation: Seven scientists search for the origins of goodness*. Princeton University Press, 2006.

[59] DUNBAR, R., AND DUNBAR, R. I. M. *Grooming, gossip, and the evolution of language*. Harvard University Press, 1998.

[60] E SANTOS, E. B., CONT, R., ET AL. The brazilian interbank network structure and systemic risk. Tech. rep., Central Bank of Brazil, Research Department, 2010.

[61] EDMONDS, J. Matroids and the greedy algorithm. *Mathematical Programming 1*, 1 (1971), 127–136.

[62] EPPSTEIN, D. Representing all minimum spanning trees with applications to counting and generation. Tech. Rep. 95-50, Department of Information and Computer Science, University of California, Irvine, CA 92717, December 1995.

[63] ERDÖS, P., AND RÉNYI, A. On random graphs i. *Publicationes Mathematicae Debrecen 6* (1959), 290.

[64] ESTRADA, E., AND BENZI, M. Are Social Networks Really Balanced? *ArXiv e-prints* (June 2014).

[65] ESTRADA, E., AND BENZI, M. Walk-based measure of balance in signed networks: Detecting lack of balance in social networks. *Physical Review E 90* (Oct 2014), 042802.

[66] ESTRADA, E., AND KNIGHT, P. K. *A First Course in Network Theory*. Oxford University Press, Oxford, 2015.

[67] EULER, L. Solutio problematis ad geometriam situs pertinentis. *Commentarii Academiae Scientiarum Imperialis Petropolitanae 8* (1736), 128–140.

[68] EXCHANGE, C. B. O. The cboe volatility index–vix. white paper, 2009.

[69] EXCOFFIER, L., AND SMOUSE, P. E. Using allele frequencies and geographic subdivision to reconstruct gene trees within a species: molecular variance parsimony. *Genetics 136* (1994), 343–359.

[70] FACCHETTI, G., IACONO, G., AND ALTAFINI, C. Computing global structural balance in large-scale signed social networks. *Proceedings of the National Academy of Sciences 108*, 52 (2011), 20953–20958.

[71] FAMA, E. F. Efficient capital markets: Ii. *The Journal of Finance 46*, 5 (1991), 1575–1617.

[72] FEHR, E., AND FISCHBACHER, U. The nature of human altruism. *Nature 425*, 6960 (2003), 785.

[73] FEIL, E. J., LI, B. C., AANENSEN, D. M., HANAGE, W. P., AND SPRATT, B. G. eBURST: Inferring patterns of evolutionary descent among clusters of related bacterial genotypes from multilocus sequence typing data. *J. Bacteriol. 186* (2004), 1518–1530.

[74] FELSENSTEIN, J. *Inferring phylogenies*. Sinauer Associates, Sunderland, Massachusetts, 2004.

[75] FRANCISCO, A. P., BUGALHO, M., RAMIREZ, M., AND CARRIÇO, J. Global optimal eBURST analysis of multilocus typing data using a graphic matroid approach. *BMC Bioinformatics 10* (2009), 152.

[76] FRANCISCO, A. P., VAZ, C., MONTEIRO, P. T., MELO-CRISTINO, J., RAMIREZ, M., AND CARRIÇO, J. PHYLOViZ: phylogenetic inference and data visualization for sequence based typing methods. *BMC Bioinformatics 13*, 1 (2012), 87.

[77] FRASER, C., ALM, E. J., POLZ, M. F., SPRATT, B. G., AND HANAGE, W. P. The bacterial species challenge: Making sense of genetic and ecological diversity. *Science 323*, 5915 (2009), 741–746.

[78] FRASER, C., HANAGE, W. P., AND SPRATT, B. G. Neutral microepidemic evolution of bacterial pathogens. *Proceedings of the National Academy of Sciences USA 102*, 6 (2005), 1968–1973.

[79] FRASER, C., HANAGE, W. P., AND SPRATT, B. G. Recombination and the nature of bacterial speciation. *Science 315*, 5811 (2007), 476–480.

[80] FREEMAN, L. C. A set of measures of centrality based upon betweeness. *Sociometry 40*, 1 (1977), 35–41.

[81] GAO, J., BULDYREV, S. V., STANLEY, H. E., AND HAVLIN, S. Networks formed from interdependent networks. *Nature physics 8*, 1 (2012), 40.

[82] GAVRIL, F. Generating the maximum spanning trees of a weighted graph. *J. Algorithms 8*, 4 (1987), 592–597.

[83] GEORGE, A., AND NG, E. On the complexity of sparse QR and LU factorization of finite-element matrices. *SIAM J. Sci. Comput. 9*, 5 (1988), 849–861.

[84] GILBERT, E. N. Random graphs. *Ann. Math. Statist. 30*, 4 (12 1959), 1141–1144.

[85] GINTIS, H. *Game theory evolving: A problem-centered introduction to modeling strategic behavior*. Princeton university press, 2000.

[86] GIRVAN, M., AND NEWMAN, M. E. Community structure in social and biological networks. *Proceedings of the National Academy of Sciences USA 99*, 12 (2002), 7821–7826.

[87] GISCARD, P.-L., ROCHET, P., AND WILSON, R. C. Evaluating balance on social networks from their simple cycles. *Journal of Complex Networks 5*, 5 (2017), 750–775.

[88] GRADY, D., THIEMANN, C., AND BROCKMANN, D. Robust classification of salient links in complex networks. *Nature Communications 3* (2012), 864.

[89] GRIFFIN, K. Testimony to the house committee on oversight and government reform. *Prepared for the US House of Representatives, Committee on Oversight and Government Reform, Hearing on Hedge Funds, Washington, DC, November 13* (2008).

[90] GROSS, T., AND BLASIUS, B. Adaptive coevolutionary networks: a review. *J. R. Soc. Interface 5*, 20 (2008), 259–271.

[91] HANAGE, W. P., SPRATT, B. G., TURNER, K. M. E., AND FRASER, C. Modelling bacterial speciation. *Philosophical Transactions of the Royal Society of London B: Biological Sciences 361*, 1475 (2006), 2039–2044.

[92] HARARY, F. On the notion of balance of a signed graph. *Michigan Math. J. 2*, 2 (1953), 143–146.

[93] HARARY, F. On the measurement of structural balance. *Behavioral Science 4*, 4 (1959), 316–323.

[94] HARDIN, G. The tragedy of the commons. *Science 162*, 3859 (1968), 1243–1248.

[95] HARRIS, J., HIRST, J. L., AND MOSSINGHOFF, M. *Combinatorics and Graph Theory*. Springer, 2008.

[96] HAUERT, C., AND DOEBELI, M. Spatial structure often inhibits the evolution of cooperation in the snowdrift game. *Nature 428*, 6983 (2004), 643.

[97] HE, X., DU, H., CAI, M., AND FELDMAN, M. W. The evolution of cooperation in signed networks under the impact of structural balance. *PloS ONE 13*, 10 (2018), e0205084.

[98] HOFBAUER, J., AND SIGMUND, K. *Evolutionary games and population dynamics*. Cambridge university press, 1998.

[99] HOLME, P., AND SARAMÄKI, J. Temporal networks. *Physics Reports 519*, 3 (2012), 97–125.

[100] HUMMON, N. P., AND DOREIAN, P. Some dynamics of social balance processes: bringing heider back into balance theory. *Social Networks 25*, 1 (2003), 17 – 49.

[101] JIANG, L., XU, Q., OUYANG, B., LANG, Y., DAI, Y., AND TONG, J. Epidemic spreading in interdependent networks. *Mathematical Problems in Engineering 2018* (2018).

[102] KARP, R. M. Reducibility among combinatorial problems. In *Complexity of computer computations*. Springer, 1972, pp. 85–103.

[103] KASNER, E., AND NEWMAN, J. R. *Mathematics and the Imagination*. Courier Dover Publications, 2001.

[104] KENETT, D. Y., GAO, J., HUANG, X., SHAO, S., VODENSKA, I., BULDYREV, S. V., PAUL, G., STANLEY, H. E., AND HAVLIN, S. Network of interdependent networks: overview of theory and applications. In *Networks of Networks: The Last Frontier of Complexity*. Springer, 2014, pp. 3–36.

[105] KERR, B., RILEY, M. A., FELDMAN, M. W., AND BOHANNAN, B. J. Local dispersal promotes biodiversity in a real-life game of rock–paper–scissors. *Nature 418*, 6894 (2002), 171.

[106] KIMURA, M. Evolutionary rate at the molecular level. *Nature 217* (1968), 624–626.

[107] KIRCHHOFF, G. Ueber die auflösung der gleichungen, auf welche man bei der untersuchung der linearen vertheilung galvanischer ströme geführt wird. *Annalen der Physik 148*, 12 (1847), 497–508.

[108] KIRKLEY, A., CANTWELL, G. T., AND NEWMAN, M. Balance in signed networks. *arXiv preprint arXiv:1809.05140* (2018).

[109] KIVELÄ, M., ARENAS, A., BARTHELEMY, M., GLEESON, J. P., MORENO, Y., AND PORTER, M. A. Multilayer networks. *Journal of Complex Networks 2*, 3 (2014), 203–271.

[110] KONG, Y., MA, J. H., WARREN, K., TSANG, R. S., LOW, D. E., JAMIESON, F. B., ALEXANDER, D. C., AND HAO, W. Homologous recombination drives both sequence diversity and gene content variation in *Neisseria meningitidis*. *Genome biology and evolution 5*, 9 (2013), 1611–1627.

[111] KOSSINETS, G., AND WATTS, D. J. Empirical analysis of an evolving social network. *Science 311*, 5757 (2006), 88–90.

[112] KRUSKAL, J. B. On the shortest spanning subtree of a graph and the traveling salesman problem. *Proc. Amer. Math. Soc. 7* (1956), 48–50.

[113] LEONIDOV, A., AND RUMYANTSEV, E. Russian interbank networks: main characteristics and stability with respect to contagion. *arXiv preprint arXiv:1210.3814* (2012).

[114] LESKOVEC, J., HUTTENLOCHER, D., AND KLEINBERG, J. Predicting positive and negative links in online social networks. In *Proceedings of the 19th International Conference on World Wide Web* (New York, NY, USA, 2010), WWW '10, ACM, pp. 641–650.

[115] LESKOVEC, J., HUTTENLOCHER, D., AND KLEINBERG, J. Signed networks in social media. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2010), CHI '10, ACM, pp. 1361–1370.

[116] LESKOVEC, J., RAJARAMAN, A., AND ULLMAN, J. D. *Mining of massive datasets*. Cambridge university press, 2014.

[117] LEVIN, S. A. Multiple scales and the maintenance of biodiversity. *Ecosystems 3*, 6 (2000), 498–506.

[118] LEWIN, M. A generalization of the matrix-tree theorem. *Mathematische Zeitschrift 181* (1982), 55–70.

[119] LI, H., AND ZHANG, Z. Kirchhoff index as a measure of edge centrality in weighted networks: Nearly linear time algorithms. In *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms* (Philadelphia, PA, USA, 2018), SODA '18, Society for Industrial and Applied Mathematics, pp. 2377–2396.

[120] LI, X., JUSUP, M., WANG, Z., LI, H., SHI, L., PODOBNIK, B., STANLEY, H. E., HAVLIN, S., AND BOCCALETTI, S. Punishment diminishes the benefits of network reciprocity in social dilemma experiments. *Proceedings of the National Academy of Sciences USA 115*, 1 (2018), 30–35.

[121] LIN, J., AND DYER, C. *Data-Intensive Text Processing with MapReduce*. Morgan and Claypool Publishers, 2010.

[122] LO, A. *Adaptive Markets: Financial Evolution at the Speed of Thought*. Princeton University Press, 2017.

[123] LO, A. W. The adaptive markets hypothesis. *The Journal of Portfolio Management 30*, 5 (2004), 15–29.

[124] MACY, M. W., AND FLACHE, A. Learning dynamics in social dilemmas. *Proceedings of the National Academy of Sciences USA 99*, suppl 3 (2002), 7229–7236.

[125] MAIDEN, M., BYGRAVES, J. A., FEIL, E., MORELLI, G., RUSSELL, J. E., URWIN, R., ZHANG, Q., ZHOU, J., ZURTH, K., CAUGANT, D. A., FEAVERS, I. M., A, M., AND SPRATT, B. G. Multilocus sequence typing: a portable approach to the identification of clones within populations of pathogenic microorganisms. *Proc. Natl. Acad. Sci. U.S.A. 95* (1998), 3140–3145.

[126] MAVROFORAKIS, C., GARCIA-LEBRON, R., KOUTIS, I., AND TERZI, E. Spanning edge centrality: Large-scale computation and applications. In *Proceedings of the 24th International Conference on*

World Wide Web (Republic and Canton of Geneva, Switzerland, 2015), WWW '15, International World Wide Web Conferences Steering Committee, pp. 732–742.

[127] MILGRAM, S. The small-world problem. *Psychology Today 1*, 1 (1967).

[128] MORONE, F., AND MAKSE, H. A. Influence maximization in complex networks through optimal percolation. *Nature 524* (2015), 65–68.

[129] MUZZI, A., AND DONATI, C. Population genetics and evolution of the pan-genome of *streptococcus pneumoniae*. *International Journal of Medical Microbiology 301*, 8 (2011), 619–622.

[130] NARENDRA, K. S., AND THATHACHAR, M. A. *Learning automata: an introduction*. Courier Corporation, 2012.

[131] NEWMAN, M. E. J. The spread of epidemic disease on networks. *Phys. Rev. E. 66* (2002).

[132] NOWAK, M. A. *Evolutionary dynamics*. Harvard University Press, 2006.

[133] NOWAK, M. A., AND MAY, R. M. Evolutionary games and spatial chaos. *Nature 359*, 6398 (1992), 826.

[134] OCHMAN, H., LAWRENCE, J. G., AND GROISMAN, E. A. Lateral gene transfer and the nature of bacterial innovation. *Nature 405* (2000), 299–304.

[135] ONNELA, J.-P., KASKI, K., AND KERTÉSZ, J. Clustering and information in correlation based financial networks. *The European Physical Journal B 38*, 2 (2004), 353–362.

[136] ONNELA, J. P., SARAMAKI, J., HYVONEN, J., SZABO, G., LAZER, D., KASKI, K., KERTESZ, J., AND BARABASI, A.-L. Structure and tie strengths in mobile communication networks. *Proc. Natl. Acad. Sci. USA 104*, 18 (2007), 7332–7336.

[137] ONNELA, J.-P., SARAMÄKI, J., KERTÉSZ, J., AND KASKI, K. Intensity and coherence of motifs in weighted complex networks. *Physical Review E 71*, 6 (2005), 065103.

[138] PAGE, L., BRIN, S., MOTWANI, R., AND WINOGRAD, T. The PageRank Citation Ranking: Bringing Order to the Web. Technical Report 1999-66, Stanford InfoLab, 1999.

[139] PAPADIMITRIOU, C. H., AND STEIGLITZ, K. *Combinatorial Optimization*. Dover, 1998.

[140] PASTOR-SATORRAS, R., CASTELLANO, C., VAN MIEGHEM, P., AND VESPIGNANI, A. Epidemic processes in complex networks. *Rev. Mod. Phys. 87* (Aug 2015), 925–979.

[141] PEARSON, T., GIFFARD, P., BECKSTROM-STERNBERG, S., AUERBACH, R., HORNSTRA, H., TUANYOK, A., PRICE, E. P., GLASS, M. B., LEADEM, B., BECKSTROM-STERNBERG, J. S., ET AL.

Phylogeographic reconstruction of a bacterial species with high levels of lateral gene transfer. *BMC Biology 7*, 1 (2009), 78.

[142] PINHEIRO, F. L., PACHECO, J. M., AND SANTOS, F. C. From local to global dilemmas in social networks. *PloS ONE 7*, 2 (2012), e32114.

[143] PINHEIRO, F. L., SANTOS, M. D., SANTOS, F. C., AND PACHECO, J. M. Origin of peer influence in social networks. *Phys. Rev. Lett. 112* (Mar 2014), 098702.

[144] PRESS, W. H., TEUKOLSKY, S. A., VETTERLING, W. T., AND FLANNERY, B. P. *Numerical recipes in C: the art of scientific computing, 1992*. Cambridge University Press, 1992.

[145] QI, X., FULLER, E., LUO, R., AND ZHANG, C. A novel centrality method for weighted networks based on the kirchhoff polynomial. *Pattern Recognition Letters 58*, C (June 2015), 51–60.

[146] RAND, D. G., AND NOWAK, M. A. Human cooperation. *Trends in Cognitive Sciences 17*, 8 (2013), 413–425.

[147] RAND, D. G., NOWAK, M. A., FOWLER, J. H., AND CHRISTAKIS, N. A. Static network structure can stabilize human cooperation. *Proceedings of the National Academy of Sciences 111*, 48 (2014), 17093–17098.

[148] RIBEIRO, P., AND SILVA, F. G-tries: an efficient data structure for discovering network motifs. In *Proceedings of the 2010 ACM Symposium on Applied Computing* (2010), ACM, pp. 1559–1566.

[149] RIBEIRO, P., AND SILVA, F. G-tries: a data structure for storing and finding subgraphs. *Data Mining and Knowledge Discovery 28*, 2 (Mar 2014), 337–377.

[150] RIJT, A. V. D. The micro-macro link for the theory of structural balance. *The Journal of Mathematical Sociology 35*, 1-3 (2011), 94–113.

[151] RINALDI, S. M., PEERENBOOM, J. P., AND KELLY, T. K. Identifying, understanding, and analyzing critical infrastructure interdependencies. *IEEE Control Systems 21*, 6 (2001), 11–25.

[152] ROBINSON, D. A., FALUSH, D., AND FEIL, E. J. *Bacterial population genetics in infectious disease*. John Wiley & Sons, 2010.

[153] RUSSO, L., TEIXEIRA, A. S., AND FRANCISCO, A. P. Linking and cutting spanning trees. *Algorithms 11*, 4 (2018), 53.

[154] SALIPANTE, S. J., AND HALL, B. G. Inadequacies of minimum spanning trees in molecular epidemiology. *J. Clin. Microbiol. 49* (2011), 3568–3575.

[155] SANTOS, F., RODRIGUES, J., AND PACHECO, J. Graph topology plays a determinant role in the evolution of cooperation. *Proceedings of the Royal Society B: Biological Sciences 273*, 1582 (2005), 51–55.

[156] SANTOS, F. C., AND PACHECO, J. M. Risk of collective failure provides an escape from the tragedy of the commons. *Proceedings of the National Academy of Sciences USA 108*, 26 (2011), 10421–10425.

[157] SANTOS, F. C., PACHECO, J. M., AND LENAERTS, T. Cooperation prevails when individuals adjust their social ties. *PLOS Computational Biology 2*, 10 (10 2006), 1–8.

[158] SANTOS, F. C., PACHECO, J. M., AND LENAERTS, T. Evolutionary dynamics of social dilemmas in structured heterogeneous populations. *Proceedings of the National Academy of Sciences 103*, 9 (2006), 3490–3494.

[159] SANTOS, F. C., SANTOS, M. D., AND PACHECO, J. M. Social diversity promotes the emergence of cooperation in public goods games. *Nature 454*, 7201 (2008), 213–216.

[160] SANTOS, F. P., PACHECO, J. M., PAIVA, A., AND SANTOS, F. C. Structural power and the evolution of collective fairness in social networks. *PloS ONE 12*, 4 (2017), e0175687.

[161] SANTOS, F. P., SANTOS, F. C., AND PAIVA, A. The evolutionary perks of being irrational. In *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems* (2015), International Foundation for Autonomous Agents and Multiagent Systems, pp. 1847–1848.

[162] SIGMUND, K. *The calculus of selfishness*. Princeton University Press, 2010.

[163] SIMPSON, E. H. Measurement of diversity. *Nature 163*, 4148 (1949), 688.

[164] SKYRMS, B. *The Stag Hunt and the Evolution of Social Structure*. Cambridge University Press, 2012.

[165] SMITH, J. M., FEIL, E. J., SMITH, N. H., ET AL. Population structure and evolutionary dynamics of pathogenic bacteria. *Bioessays 22*, 12 (2000), 1115–1122.

[166] SNEATH, P. H. A., AND SOKAL, R. R. *Numerical taxonomy; the principles and practice of numerical classification*. W. H. Freeman, San Francisco, 1973.

[167] SOUTO, P. Marketopolis: A market simulation based on investors decisions. Master's thesis, Instituto Superior Técnico, 2016.

[168] SOUTO, P. C., TEIXEIRA, A. S., FRANCISCO, A. P., AND SANTOS, F. C. Capturing financial volatility through simple network measures. In *International Workshop on Complex Networks and their Applications* (2018), Springer, pp. 534–546.

[169] SPRATT, B. G., HANAGE, W. P., AND FEIL, E. J. The relative contributions of recombination and point mutation to the diversification of bacterial clones. *Curr Opin Microbiol. 4*, 5 (2001), 602—-606.

[170] STUMPF, M. P., WIUF, C., AND MAY, R. M. Subnets of scale-free networks are not scale-free: sampling properties of networks. *Proceedings of the National Academy of Sciences USA 102*, 12 (2005), 4221–4224.

[171] SUTTON, R. S., AND BARTO, A. G. *Reinforcement learning: An introduction*. Cambridge, MA: MIT Press, 2011.

[172] SZABÓ, G., AND FÁTH, G. Evolutionary games on graphs. *Physics Reports 446*, 4 (2007), 97 – 216.

[173] TEIXEIRA, A. S., MONTEIRO, P. T., CARRIÇO, J. A., RAMIREZ, M., AND FRANCISCO, A. P. Spanning edge betweenness. In *Workshop on mining and learning with graphs* (2013), vol. 24, pp. 27–31.

[174] TEIXEIRA, A. S., MONTEIRO, P. T., CARRIÇO, J. A., RAMIREZ, M., AND FRANCISCO, A. P. Not seeing the forest for the trees: Size of the minimum spanning trees (msts) forest and branch significance in mst-based phylogenetic analysis. *PLOS ONE 10*, 3 (03 2015), 1–15.

[175] TEIXEIRA, A. S., MONTEIRO, P. T., CARRIÇO, J. A., SANTOS, F. C., AND FRANCISCO, A. P. Using spark and graphx to parallelize large-scale simulations of bacterial populations over host contact networks. In *International Conference on Algorithms and Architectures for Parallel Processing* (2017), Springer, pp. 591–600.

[176] TEIXEIRA, A. S., MONTEIRO, P. T., CARRIÇO, J. A., SANTOS, F. C., AND FRANCISCO, A. P. Large-scale simulations of bacterial populations over complex networks. *Journal of Computational Biology 25*, 8 (2018), 850–861.

[177] TEIXEIRA, A. S., SANTOS, F. C., AND FRANCISCO, A. P. Spanning edge betweenness in practice. In *Complex Networks VII: Proceedings of the 7th Workshop on Complex Networks CompleNet 2016* (2016), Springer International Publishing, pp. 3–10.

[178] TEIXEIRA, A. S., SANTOS, F. C., AND FRANCISCO, A. P. Emergence of social balance in signed networks. In *Workshop on Complex Networks CompleNet* (2017), Springer, pp. 185–192.

[179] TEIXEIRA, A. S., SANTOS, F. C., FRANCISCO, A. P., AND SANTOS, F. P. Fairness in multiplayer ultimatum games through degree-based role assignment. In *Book of Abstracts* (2018), International Workshop on Complex Networks and their Applications, pp. 218–220.

[180] TRAN, T. D., HOFRICHTER, J., AND JOST, J. An introduction to the mathematical structure of the Wright-Fisher model of population genetics. *Theory in Biosciences 132*, 2 (2013), 73–82.

[181] TRAVERS, J., AND MILGRAM, S. An experimental study of the small world problem. *Sociometry 32*, 4 (1969), 425–443.

[182] TUTTE, W. T. Lectures on matroids. *J. Res. Nat. Bur. Standards Sect. B 69* (1965), 1–47.

[183] V LATORA, V. N., AND RUSSO, G. *Complex Networks: Principles, Methods and Applications*. Cambridge University Press, Cambridge, 2018.

[184] VAN BOECKEL, T. P., BROWER, C., GILBERT, M., GRENFELL, B. T., LEVIN, S. A., ROBINSON, T. P., TEILLANT, A., AND LAXMINARAYAN, R. Global trends in antimicrobial use in food animals. *Proceedings of the National Academy of Sciences 112*, 18 (2015), 5649–5654.

[185] VAN SEGBROECK, S., DE JONG, S., NOWÉ, A., SANTOS, F. C., AND LENAERTS, T. Learning to coordinate in complex networks. *Adaptive Behavior 18*, 5 (2010), 416–427.

[186] VASCONCELOS, V. V., SANTOS, F. C., PACHECO, J. M., AND LEVIN, S. A. Climate policies under wealth inequality. *Proceedings of the National Academy of Sciences USA 111*, 6 (2014), 2212–2216.

[187] VASCONCELOS, V. V., SANTOS, F. P., SANTOS, F. C., AND PACHECO, J. M. Stochastic dynamics through hierarchically embedded markov chains. *Physical Review Letters 118*, 5 (2017), 058301.

[188] VESPIGNANI, A. Complex networks: The fragility of interdependency. *Nature 464*, 7291 (2010), 984.

[189] VISENTIN, G., BATTISTON, S., AND D'ERRICO, M. Rethinking financial contagion. *arXiv e-prints* (2016).

[190] WATTS, D. J., AND STROGATZ, S. H. Collective dynamics of small-world networks. *Nature 393* (1998), 440–442.

[191] WHITNEY, H. On the abstract properties of linear dependence. *American Journal of Mathematics 57*, 3 (1935), 509–533.

[192] WILSON, D. J., GABRIEL, E., LEATHERBARROW, A. J., CHEESBROUGH, J., GEE, S., BOLTON, E., FOX, A., HART, C. A., DIGGLE, P. J., AND FEARNHEAD, P. Rapid evolution and the importance of recombination to the gastroenteric pathogen *Campylobacter jejuni*. *Molecular Biology and Evolution 26*, 2 (2009), 385–397.

[193] WYMAN, O. Managing complexity: The state of the financial services industry 2015, 2015.

[194] XIN, R. S., GONZALEZ, J. E., FRANKLIN, M. J., AND STOICA, I. Graphx: A resilient distributed graph system on spark. In *First International Workshop on Graph Data Management Experiences and Systems* (2013), GRADES '13, ACM, pp. 2:1–2:6.

[195] Y QIAN, Y LI, M. Z. G. M., AND LU, F. Quantifying edge significance on maintaining global connectivity. *Scientific Reports 7*, 45380 (2017).

[196] ZAHARIA, M., CHOWDHURY, M., DAS, T., DAVE, A., MA, J., MCCAULEY, M., FRANKLIN, M. J., SHENKER, S., AND STOICA, I. Resilient distributed datasets: A fault-tolerant abstraction for in-memory cluster computing. In *Proceedings of the 9th USENIX Conference on Networked Systems Design and Implementation* (2012), NSDI'12, USENIX Association, pp. 2–2.

[197] ZAHARIA, M., CHOWDHURY, M., FRANKLIN, M. J., SHENKER, S., AND STOICA, I. Spark: Cluster computing with working sets. In *Proceedings of the 2Nd USENIX Conference on Hot Topics in Cloud Computing* (2010), HotCloud'10, USENIX Association, pp. 10–10.

[198] ZHU, X., AND GHAHRAMANI, Z. Learning from labeled and unlabeled data with label propagation. Technical report cmu-cald-02-107, Carnegie Mellon University, 2002.