Leader-follower consensus under peer-pressure in complex networks

Eusebio Vargas-Estrada

Department of Mathematics

University of Strathclyde

Glasgow, UK

September 2015

This thesis is submitted to the University of Strathclyde for the degree of Doctor of Philosophy in the Faculty of Science.

Acknowledgments

I came to Glasgow looking for an opportunity of studying and pursuing a degree. At the end of this path I have had more than that: I have found friends. This whole adventure and exciting times, full of discoveries, knowledge, mistakes, laughs, frustrations and successes, would never have been possible if my supervisor would not have accepted my application to be part of his laboratory: my very honest gratitude to Professor Ernesto Estrada, who gave me this opportunity. He has guided my steps during these years through the complexities of being a PhD student. I have learned so much from him as a researcher, but most important, I also have learned from him as a human being. Many thanks Ernesto.

I would like to take the opportunity to thank the National Council for Science and Technology of Mexico (CONACYT) for the scholarship provided to support my studies abroad and achieve this goal in my career. I also would like to thank Dr. Thomas W. Valente for sharing his data sets of social networks used for our simulations. Likewise, thank you to the members of the academic and technical staff of the Department of Mathematics and Statistics at the University of Strathclyde for their help.

During my time in Glasgow I had the privilege of knowing people who helped me in one way or another, many of them remain as good friends, but I specially would like to thank Puri, for her support, friendship and kind help.

Of course, an special thanks to my parents Eusebio and Alicia, for their unconditional love and support during my whole life. A deep thank you goes also to my grand parents, Gabriel and Elena, because my journey to Glasgow started with them many many years ago, when I was a child.

From the bottom of my heart, a very special thank you to my beloved wife Marita. She came with me convinced that I could complete this adventure. She has always trusted me, and her support and love have made this time here the best of my life. We have found each other during these years. She is the reason I got up when I thought I had nothing else to give. I admire her strength and constancy.

At the end of this journey, there is a little person that joined us: our beloved son Gabriel, a small piece of great motivation and joy for our life. He is our blessing and our best award.

Eusebio Vargas Estrada, Glasgow, U.K., 2015.

Abstract

Synchronisation is an important process for different kinds of systems, such as biological, chemical, physical and social. Among the related synchronisation problems, consensus has received high attention because of the distributed properties shown by its models and the possibility they offer for controlling complex systems. When dealing with consensus processes in social networks, we known from empirical evidence that the formation of opinions is not free from being influenced by people around every actor, and more, it is well known that some of the actors may play a leading role and guide a social system to a final state different from the pure average consensus. A main paradigm while modelling interactions among actors in social networks is that every actor receives and transmits information from and to her nearest neighbours, thus implicitly assuming that the decisions of a given actor only are influenced by their directly connected peers, and not taking into account indirect influences coming from not directly connected peers in the same social network, for example, the influence coming from the friend's friend of a friend.

Our work studies consensus processes in the presence of influence coming from not only those directly connected actors, but from other ones in the same network. We call this influence peer pressure (PP). We propose a consensus model that takes into account direct and indirect PP modelled as a function of the social distance among actors. We apply this consensus model to different real social networks assuming three different decay laws for the strength of the PP, and in the presence of leaders and without them. We choose those nodes acting as leaders according to different centrality criteria, as well as randomly, and compare their performance for driving the system. Since it is natural that different leaders may diverge in their positions, we introduce a divergence parameter among the initial states of the

leaders with respect to the average consensus of the system, to take this feature into account in our model. We then analyse the effects of PP on two different real cases of diffusion of innovations processes.

We show that as the strength of indirect PP increases, the centrality criteria used to select the leaders has a decaying effect on the effectiveness of such leaders to better drive a consensus process, allowing random leaders to be as good as those with better centrality. Our work also shows that, despite divergence among leaders induces higher times for reaching consensus, this effect is reduced for stronger levels of PP present in the system. For the case of diffusion of innovations our model reproduces the behaviour of the empirical data, and we demonstrate that certain levels of PP are necessary to match the results coming from two different studies, supporting our hypothesis that indirect PP is an important factor to be taken into account when modelling opinion formations in social networks.

Leaders emerging by global centrality criteria in networks with tightly connected groups can be counterproductive. This can be tackled by selecting node-leaders in a local basis. This effect is also reduced when indirect PP is allowed to be higher. This finding points to the fact that distance among nodes is an important characteristic for consensus processes. For the purpose of studying this structural feature, we propose a distance-sum heterogeneity index based on a fictional consensus process. We conjecture that an special type of graph, that we call *complete split graph*, is related with the maximization of the index, and based on this conjecture we study the relative distance-sum heterogeneity of random graphs and different real-world networks, which allows us to characterise them. We propose a spectral representation of the distance-sum heterogeneity index for networks that we call S-plots. We also study the relation between the time for consensus and the distance-sum heterogeneities in complex networks from different nature.

Contents

List of Figures

List of Tables

INTRODUCTION

There is a concept that pervades our daily lives and is changing the way people approach science, medicine, economics, business, and even the study of intelligence in natural and artificial systems: complexity. Diverse real-life phenomena can be represented as complex systems models that reproduce, with certain amount of success, situations like cellular interactions, chemical reaction-diffusion systems, the dynamics of stock markets, motorway traffic, neuronal interactions, the formation of opinions in social systems, power grids flows, the dynamics of the internet, air lane transportation, street plans, among many others.

But, what complex systems are? They are systems that comprise many interconnected parts with the ability to interact and generate a new quality of collective behaviour through self-organization, i.e., the spontaneous formation of temporal, spatial or functional structures [1]. Complex systems are much more than a sum of their parts, they are often characterised as having extreme sensitivity to initial conditions as well as emergent behaviours that are not readily predictable or even completely deterministic [2].

The recent advances in technology allow collecting and processing great amounts of data from different systems and have facilitated the ability to synthesize networked systems that resemble, at some point, their natural counterparts in terms of their functional and operational complexity [3]. Since we live in a complex interconnected world, it has become very important to understand more deeply the role that those inter-part interactions play in the collective functionality of the systems and in the emergence of different global properties. "There are people studying the nature of the parts or components of the systems, and there are also people interested in the nature of the connections, but there is a group of people that studies the aspect of the pattern of connections between components (the structure) which

is fundamental to the behaviour of the whole system" [4]. *Network Science* has become a powerful framework for modelling, analysing, simulating and designing such complex systems. A *network* can be defined as a diagrammatic representation of a system consisting of nodes or vertices, which represent the entities of the system. These nodes are joined by links or edges that represent a particular kind of interconnection among them [5]. Networks are graphs that represent something, they are a representation or model of observable reality. A central idea in this field is: the structure and attributes of the network influence its dynamical properties.

When we abstract away the complex interaction geometries of a system associated with the actors or agents by representing them as nodes and encoding their interactions as edges the advantage of applying network science methods for the analysis and synthesis of these systems is evident: the set of tools from network science provides a handful way to examine how the structure of the underlying interaction topology among agents disembogue to distinct global behaviour. A graph-based abstraction of a complex networked system contains high-level descriptions of the networked topology in terms of nodes and edges, even though it sacrifices some information about what exactly is shared among the elements of the system.

Complex networks are the skeleton of complex systems and their language is the language of graph theory [5], thus in order to get the most of them we need to go through all the vast machinery of this mathematical area that in combination with the theory of dynamical systems build a powerful framework from which many questions pertaining complex systems can be answered.

The common thread of these systems is the presence of an underlying network that has influence on the behaviour of the complete system, so we can think that such networked systems present some similar characteristics that make them suitable for being modelled by network science:

- They consist of dynamic units.
- The dynamic units posses decision making capabilities.
- The units can receive and transmit information among themselves.
- The system presents a signal exchange network.

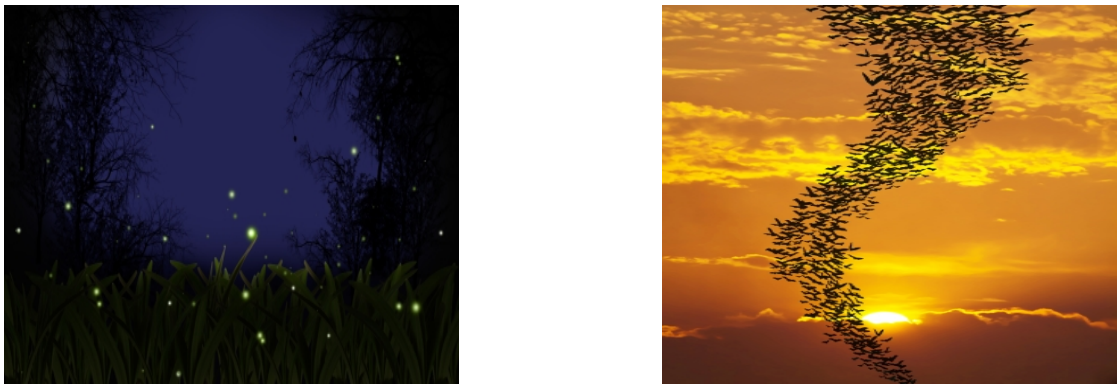
The realization of the signal exchange network depends on the kind of system and can be through wired or wireless protocols for engineering systems, chemical reactions for biological systems, or psychological and sociological interactions for social systems. The configuration of the exchange network can vary according to the characteristics of the system and can be, in general, from three different types: [3]

1. Static networks: In these kind of networks the edges are static, that means, the edge set will not be time varying. This can resemble for example a situation when a static communication network has been established and through which the information flows.

2. Dynamic, state-dependent networks: For this networks, the edge set is time varying, this characteristic allows that edges may disappear at one time, and reappear at another time, as a function of the underlying state of the network components (nodes). This might be the case of a network of mobile robots with range sensors, as they move their state might change according to their sensory range, which can make edges to appear or disappear, meaning that some robots can be within their sensor range while others not.

3. Random networks: In this special class of networks the existence of an edge is given by a probability distribution rather than some deterministic condition. A situation that can show this characteristic is in the communications settings when the quality of the communication channels can be modelled as being probabilistic in nature.

Among others, synchronisation is one of the most captivating phenomena in complex systems, it has been observed in biological, chemical, physical, engineered and social systems, where it has been shown to be an important process for their function. Visual and acoustic interactions make fireflies flash, crickets chirp, and an audience clap in synchrony (see illustrations in fig 0.0.1) [6]. One form of synchronisation is to reach an agreement on a certain state-value of the system where it is necessary for all of the nodes to calculate some function of certain parameters. When all nodes in a network calculate the same function of the initial values in the system, they are said to reach an state agreement or *consensus* [6].



(a) (b)

Figure 0.0.1: Illustrations of collective behaviour. (a) Representation of fireflies flashing in synchrony. Image courtesy of *suphakit73* at FreeDigitalPhotos.net (b) A colony of bats flying together. Image courtesy of *Exsodus* at FreeDigitalPhotos.net

In networks of agents (or dynamic systems), consensus means to reach an agreement regarding a certain quantity of interest that depends on the state of all agents. A consensus algorithm is an interaction rule that specifies the information exchange between an agent and all of its neighbours on the network [7] .

Many examples of opinion consensus are well documented across the social sciences, these include examples of behavioural flocking in popular cultural styles (clothing, hair, and music), emotional contagion, collective decision making and walking behaviour of pedestrians, among others (see fig. 0.0.2a as illustration) [8–11]. Consensus also is important in engineering systems consisting of many simple parts communicating to each other. In these systems, consensus must be reached for performing a global task, such as spacecraft alignment, multi-vehicle formation control or distributed sensor fusion [3, 8, 12]. The problem is particularly challenging because communication channels have limited range and experience fading and dropout [13], thus, it is important to design reliable and appropriate models for cooperative control on networked systems which allow them to reach consensus on shared information in the presence of limited and/or unreliable information exchange, and even dynamically changing interaction topologies [8].

The coordination of multi-agent networked systems using control and graph theoretic tools offers the possibility of steering complex systems by controlling some

of their elements [3, 7, 12, 14]. The consensus algorithm has been used to control such systems by means of a cooperative exchange of information process where the graph Laplacian plays an important role [8, 15–20]. This process has been applied under the nearest neighbour paradigm, but empirical evidence shows that long range interactions in networks can be taken into account to analyse the impact they can produce in the dynamics of complex systems [21, 22].

Networks have been used extensively in the social sciences to represent interpersonal relationships [23]. In these cases, the vertices correspond to individuals in a group or society, and the edges join pairs of individuals that are related in some way, for example, there is a link between individuals x and y if x likes, hates, agrees with, avoids, or communicates with individual y. Such representations have extended to relationships between groups of individuals, and have proved useful in a number of contexts ranging from kinship relations in certain primitive tribes to relationships between political parties [23]. The representations are called social networks, and they also have been used in political sciences to study international relations, where nodes correspond to nations or group of nations, and the edges join pair of nations that are allied, maintain diplomatic relations, agree on a particular strategy, among other reasons.

(a)

(b)

Figure 0.0.2: Illustration of social networks. (a) Protestors gathered at Victory Monument in Bangkok, Thailand. Image courtesy of *duron123* at FreeDigitalPhotos.net (b) Representation of the leader-follower situation in a group, one actor can play a key role and influence the behaviour of the rest of them. Image courtesy of *renjith krishnan* at FreeDigitalPhotos.net

In social networks, the study of information flow and sharing is of primary interest, some questions related with this area are for example, how the structure of the network may impact individuals to form their opinions, or how fast an information can be spread trough the network, or how a group of people can agree on a point of view regarding a particular situation [9, 11, 24–29]. In these situations, one or multiple actors can play a leading role during the consensus process, these actors are called *leaders* and guide the dynamics of consensus by means of the effect they produce on the rest of the group who follows them, which are called *followers* (see fig. 0.0.2b as illustration) [30]. In a leader-follower consensus process with a single leader, actors try to reach an agreement that is biased to the state of the leader, while in the case of multiple (stationary) leaders all followers converge to the convex hull formed by the leaders' states.

Our work deals with one important question: What is the role played by the combination of direct and indirect social influences on the collective decisions made by a social group? By applying network and linear dynamical systems theories, we present a model that captures the elusive concept of peer pressure (PP) exerted over a social actor to reach consensus in a complex social system. Using real-world (simple and undirected) social networks we show that the level of PP determines how fast a global consensus is reached in a social group. The leaders emerging over different levels of PP differ substantially in the position they occupy in the group. PP can also overcome the barriers imposed to a global consensus by the existence of tightly connected communities with local leaders or the existence of leaders with poor cohesiveness of opinions. Even a moderate level of PP is necessary to explain the rate at which innovations are diffused through a variety of social groups.

As distance among driver-nodes shows to be an important characteristic in consensus processes when leaders emerge in structures with communities, we analyse the distance-sum distribution on random graphs and real networks, and propose a mathematical index based on a hypothetical consensus process to study this aspect on complex networks.

We have structured the present thesis in two main parts. In part I we present the theoretical background used to build our proposals and analysis, the body of the first part is constituted of 2 chapters. In Chapter 1 we present general concepts from graph theory, networks, dynamical systems and consensus models. Chapter

2 shows a literature review that includes the main works used on this thesis, and shows the path that has helped building the actual theory on consensus protocols: from distributed algorithms, to the agreement problem and ending on the average consensus model. A brief mention about the latest works (at the time this thesis is written) on controllability of complex networks has been added at the end due to the close relation with the work presented here.

The second part of the thesis comprises the results and findings coming from our published works. Chapter 3 presents our results on consensus under peer pressure in social networks, and its impact in processes like leadership emerging and diffusion of innovations. Chapter 4 gives an analysis of some structural characteristics that play a key role when trying to reach consensus, and sets the necessity of studying the heterogeneities in distance among nodes.

In Chapter 5, we tackle the problem of studying the distance-sum heterogeneities by introducing a new index that takes into account this feature. We also correlate the new index with the times for consensus for different real networks. Finally, Chapter 6 presents our conclusion and final remarks, as well as future works that can be directly derived from our current steps. We have included at the end the numerical results in tables, figures and scropts used for our work in appendices A, B and C. All references are listed at the last part of this thesis.

Part I

Theory

Chapter 1

STRUCTURE AND DYNAMICS OF NETWORKS

Structure of Networks

In order to be able to have a *conversation* with the complex networks, we need to speak their language: *graph theory*. We use the word *network* to mean a graph that carries some numerical information which depends on the application we are dealing with and it may consists of values associated to the edges and to the vertices relevant to the application under consideration, i.e., a network conveys additional quantitative information [23]. Thus, the structural part of a network is modelled by a graph, which consists of nodes and links, and a mapping function that defines how nodes connect to each other. These characteristics give enough information to draw a diagram using dots as nodes and lines as links. Then we start by introducing, and formally defining, some concepts and properties of graphs that will be used through the rest of the chapters and that will allow us to understand what the networks are telling us about the systems they represent.

1.1 Graphs

There are many situations when we may use a group of points joined together either by lines or arrows to represent something of our interest. These points may stand for people, places or atoms, and the lines or arrows may be kindship relations, pipelines or chemical bounds. These kinds of diagrams are known under different names: sociograms (psychology), simplexes (topology), circuit diagrams (physics), organisational structures (economics), communication networks (engineering), family trees, etc. [31]

1.1.1 Sets, binary relations and graphs

To define a graph we need to use some concepts and notations from the theory of sets. We do not intend to develop all the basic concepts of sets, as this out of our scope, but only broadly introduce a few of them in order to be able to establish a brief set-based definition of graphs.

A set is a collection of objects of any nature which are called its elements or members [31]. For convenience, sets are represented by upper case letters (A, S, X) and their elements by lower case letters (a, b, x, z). A set can be defined

by enumerating its elements or by writing a property characteristic of them, a *membership* law, for example $S = \{a^2 |\, a \text{ is an integer}, 1 < a < 10\}$.

We write $a \in S$ and $b \notin S$ to indicate that a is element of S, and that b is not element of S. $|S|$ denotes the number of elements of S, also called the *order* of S. If all elements of S are also elements of T, then S is a subset of T, written $S \subseteq T$. The notation $S \subset T$ means that S is a subset of T but is nor identical to T, so that T has at least one element that is not in S.

If S and T are two sets, then $S \cup T$ means the union of S and T, the set of everything that is either a member of S or a member of T (or both), and $S \cap T$ is the intersection, the set of common elements. The set-theoretic difference $S \setminus T$, also called the relative complement of T in S, consists of all elements of S that are not members of T. The Cartesian product $S \times T$ is the set of all ordered pairs $\{a, b\}$, where a is a member of S and b is a member of T. Both union and intersection satisfy the commutative and associative laws for any sets R, S and T:

$$S \cup T = T \cap S \tag{1.1.1}$$

$$S \cup T = T \cup S \tag{1.1.2}$$

$$S \cap T = T \cap S \tag{1.1.3}$$

$$R \cup (S \cup T) = (R \cup S) \cup T \tag{1.1.4}$$

$$R \cap (S \cap T) = (R \cap S) \cap T. \tag{1.1.5}$$

To prove that two sets are equal, we often prove that every member of one set is an element of the other, and conversely [32]. This is, if we want to show that $A = B$, first we prove that $A \subseteq B$ and then prove that $B \subseteq A$. For example, if we want to prove $R \cap (S \cap T) = (R \cap S) \cap T$, first we observe that any member w of $R \cap (S \cap T)$ is both a member of R and a member of $S \cap T$, and the latter means that x belongs to both S and T. So all $w \in R$, $w \in S$ and $w \in T$ are true. From these arguments we see that both $w \in R$ and $w \in S$ are true, so are $w \in R \cap S$, and $w \in T$, therefore $w \in (R \cap S) \cap T$ and we actually have shown what we were looking for: $R \cap (S \cap T) = (R \cap S) \cap T$.

Binary relations occur frequently in mathematics and are present in everyday

life [32]. For example, the ordinary mathematical relations $<$, $=$, $>$, $\leq$ and $\geq$ are binary relations on number sets, the $\subset$ and $\subseteq$ are binary relations on collections of sets, and so forth. Then, if we let S to be the set of all living people, the statement "*is the cousin of*" is a binary relation on S. Formally, a *binary relation* $\sim$ on a set S is a rule that stipulates, given any elements w and z of S, whether w has a certain relationship with z (written as $w \sim z$) or not (written as $w \nsim z$).

We can represent any binary relation on a set S by a diagram where the elements of S are shown as points (*vertices*), and if the binary relation $w \sim z$, $w \in S$, $z \in S$ is true, then we draw a line (*edge*) from w to z with its direction indicated by an arrow at the end of the line (see fig. 1.1.1b), and provided that S is finite, then all information about any binary relation on S can be shown in a diagram. This diagram is called a *directed graph* or *digraph*. Moreover, of we have that $w \sim w$ is true for any $w \in S$, the diagram is a *looped digraph*.

Figure 1.1.1: Example of diagrams for different binary relations among a set of three elements x, y and z. (a) Graph: symmetric binary relation with no arrows (b) Digraph: non-symmetric binary relation with arrows.

The binary relation $\sim$ on S is called reflexive if $w \sim w$ is true for all $w \in S$, and antireflexive if $w \sim w$ is never true (equivalently, if $w \nsim w$ is true for all $w \in S$). Additionally, if $z \sim w$ is true whenever $w \sim z$ is true for all w, $z \in S$, then $\sim$ is called symmetric, and if the relation is symmetric, the arrows can be omitted from its diagram (see fig. 1.1.1a).Then, building from these definitions, we can state the following definition: [32]

The diagram of a symmetric, antireflexive binary relation on a finite set is called a *graph*.

As an example, suppose we have the following binary relations defined on the set $S = \{1, 2, 3, 4\}$.

$$\begin{aligned} x > y \quad & means \quad x\, is\, greater\, than, x \\ x \sim y \quad & means \quad x = y \pm 1 \\ x \approx y \quad & means \quad y = 2x \\ x \bullet y \quad & means \quad x = y^2 \end{aligned}$$

These relations have the corresponding subsets

$$\begin{aligned} S_{>} &= \{(2,1),(3,1),(3,2),(4,1),(4,2),(4,3)\} \\ S_{\sim} &= \{(1,2),(2,1),(2,3),(3,2),(3,4),(4,3)\} \\ S_{\approx} &= \{(1,2),(2,4)\} \\ S_{\bullet} &= \{(4,2),(1,1)\} \end{aligned}$$

We can represent these relations by diagrams which are shown in figure 1.1.2. The diagrams of these binary relations give us the visual version of them and constitute what we call *graphs*.

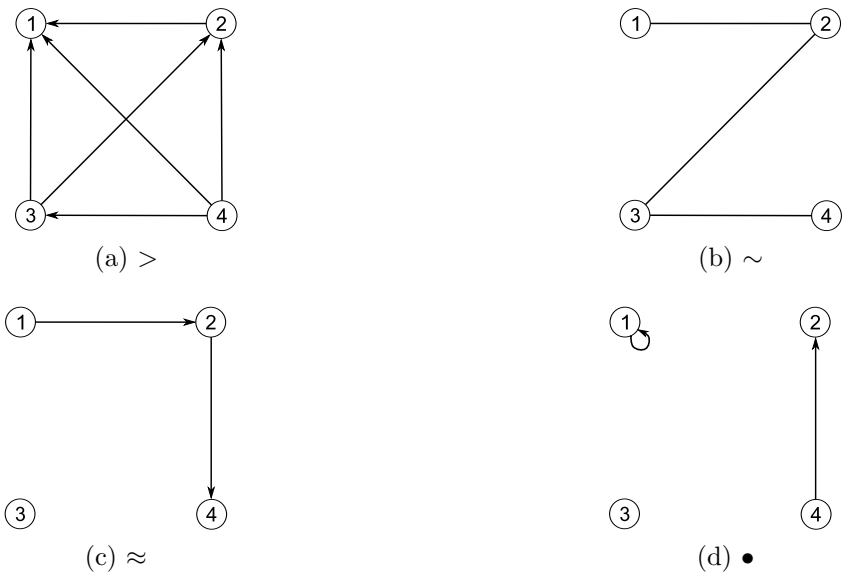


Figure 1.1.2: Diagrams of binary relations defined on the set $S = \{1, 2, 3, 4\}$. a) $x > y$ b) $x \sim y$ c) $x \approx y$ d) $x \bullet y$

1.1.2 Graph theory and networks

The theory of networks has recently witnessed a growing interest and development among different scientific sectors that have produced several works trying to condense all this knowledge into a general framework. The work from Estrada, E. (2011) [5], which we mainly follow for defining the necessary concepts from network theory to build this thesis, provides a formal, broad and deep view of the field suitable for the different scientific areas studying complex networks. It is worth to mention that due the close relation between graph theory and network theory, the words graph and network are used indistinctly in practice, and so do we trough the rest of the thesis.

Our definition of graph would not be useful if we would like to take into account the way each pair of nodes are linked together which can lead to a different types of networks (see fig. 1.1.3):

1. Simple networks: every pair of nodes are connected by a simple segment of line.
2. Directed networks: the links connecting every pair of nodes start in a given node and end in another one (*directionality*).
3. Pseudo networks: there is more than one link between a pair of nodes (*multi-links*) and even some links can join a node to itself (*self-loops*).
4. Weighted networks: the links joining the nodes have real numbers assigned to them (*weights*).

Then, a general definition of network that accounts for the last considerations is: a network is the triple $G = (V,\, E,\, f)$ where V is a finite set of vertices or nodes, $E \subseteq V \times V = \{e_1,\, e_2, ...,\, e_m\}$ is a set of edges or links and f is a mapping which associates some elements of E to a pair of elements of V, such as that if $v_i \in V$ and $v_j \in V$ then $f : e_p \to [v_i,\, v_j]$ and $f : e_q \to [v_j, v_i]$; and a weighted network is the triple $G = (V,\, W\, f)$ defined by replacing the set of edges E by a set of edges weights $W = \{w_1,\, w_2, ..., w_m\}$ such that $w_i \in \mathbb{R}$.

Now, taking as starting point our last definition of networks, we are going to define more concepts from graph and network theories that will be used trough

the thesis and that complement the necessary background for us to speak the *network-language*.

1.1.2.1 Some basic definitions

- *Adjaceny and incidence*: We say that two nodes u and v are adjacent if they are joined by a link $e = \{u, v\}$. The nodes u and v are said to be incident with the link e, and the link e is said to be incident with the nodes u and v. In a similar way, two links $e_1 = \{u, v\}$ and $e_2 = \{v, w\}$ are considered adjacent if they are both incident with at least one node.

- *Neighbourhood*: The neighbourhood $N(v_i) \subseteq V$ of the node v_i is the set $\{v_j \in V \mid v_i v_j \in E\}$ which corresponds to the set of all nodes that are adjacent to v_i.

- *Degree:* The degree of a node v is the number of of links incident with v.

- *Walk*: A (directed) walk of length k is a succession of (not necessarily different) k links of the form uv, vw, wx, ..., yz. This walk is denoted by $uvwx...yz$ and is referred to as a walk between u and z.

- *Path*: A path is a walk in which all the links and all the nodes are different, otherwise is called a trail.

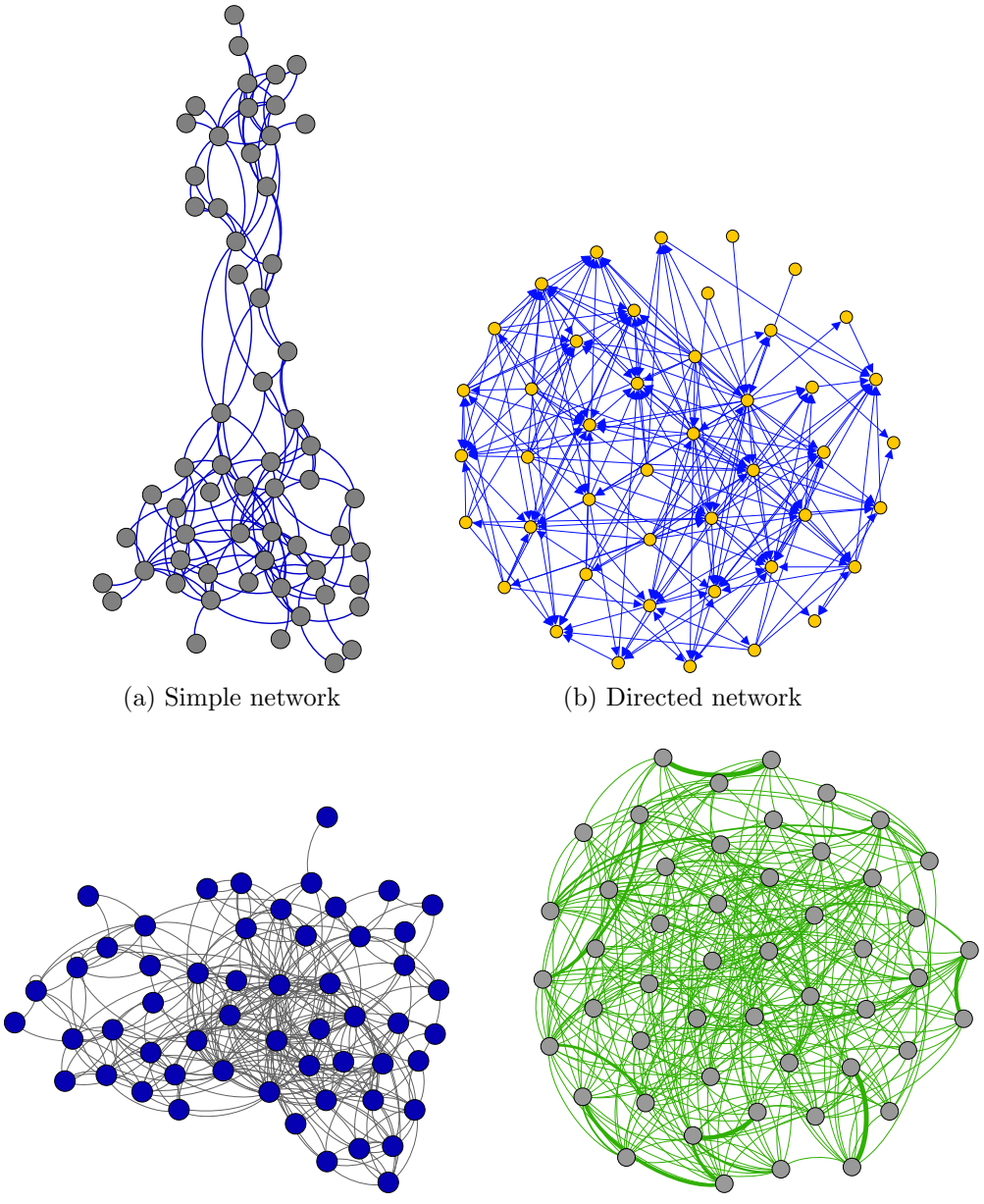
- *Connected*: A graph is connected if there is a path between each pair of nodes, and is disconnected otherwise.

- *Connected component*: Every disconnected graph can be split up into a number of connected subgraphs called components.

- *Strongly connected*: A directed graph is strongly connected if there is a directed path from u to v and a directed path from v to u for every pair of distinct nodes.

1.1.2.2 Special types of graphs

- *Regular.* A graph is regular if all its nodes have the same degree. A regular graph is r-regular (or regular of degree r) if the degree of each node is r.



(a) Simple network

(b) Directed network

(c) Pseudo network

(d) Weighted network

Figure 1.1.3: Different types of networks. (a) Community of dolphins (b) Food web St. Martin (c) Chemical elements in the atmosphere of the star Titan (d) Random weighted network. The thickness of the edges are proportional to their weight

- *Complement.* The complement of a graph G is the graph $\bar{G}$ created by taking all the nodes in G and joining two nodes whenever they are not linked in G.

- *Complete graph.* A complete graph is a graph in which each node is joined to each of the others by exactly one link. The complete graph with n nodes is denoted by K_n.

- *Null graph.* A null graph is a graph with no links. The null graph with n nodes is denoted by N_n. The null graph with n nodes is regular of degree 0.

- *Cycle graph.* A cycle graph is a graph consisting of a single cycle of vertices and edges. The cycle graph with n nodes is denoted by C_n.

- *Subgraph.* A graph $G_s = (V_s, E_s)$ is called subgraph with respect to a given graph $G = (V, E)$ if $V_s \subseteq V$ and $E_s = \{\{v_i, v_j\} \in E \,|\, v_i, v_j \in V_s\}$. In other words, the subgraph V_s consists of the vertices in the subset V_s of V and edges in G that are incident to vertices in V_s.

- *Bipartite graph.* A bipartite graph is a graph whose set of nodes can be split into two subsets V_1 and V_2 in such a way that each edge of the graph joins a node in $V_1$and a node in V_2.

- *Complete bipartite graph.* A complete bipartite graph is a bipartite graph in which each node in V_1 is joined to each node in V_2 by just one link. The complete bipartite graph with r nodes in V_1 and s nodes in V_2 is denoted by $K_{r,s}$. The graph $K_{r,s}$ is the same as $K_{s,r}$ and it has $r + s$ nodes (r node of degree s and s nodes of degree r) and rs links.

- *Trees.* A tree is a connected graph with no cycles. In this graph, there is at least a path between each pair of nodes.

- *Spanning tree.* A spanning tree if a graph is a subgraph of this that includes every node and is a tree.

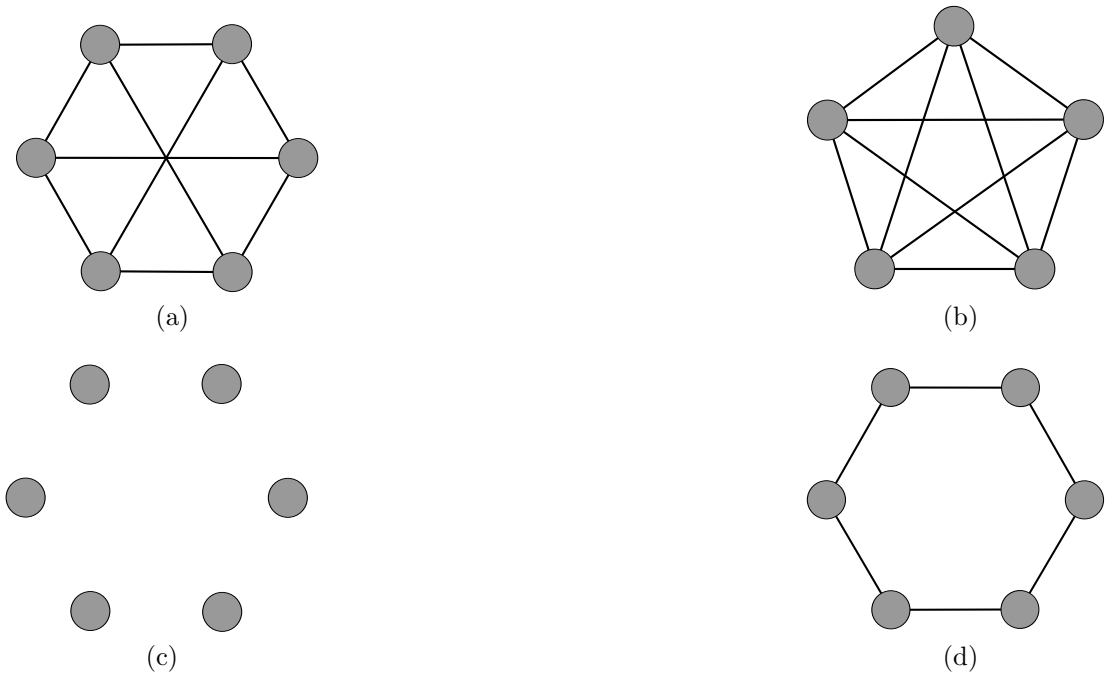


Figure 1.1.4: Examples of graphs. (a) Regular. (b) Complete. (c) Null. (d) Cycle

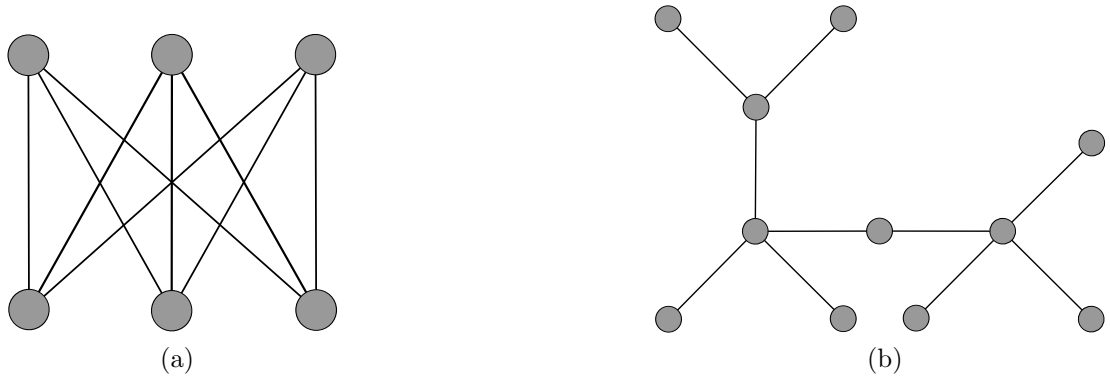


Figure 1.1.5: Examples of graphs. (a) Bipartite. (b) Tree

1.1.2.3 Structural properties of networks

Degree distributions

For any network, we can represent the degrees of its nodes as a vector which we call the *vector degree* $\boldsymbol{k}$. From this vector degree we can derive an important

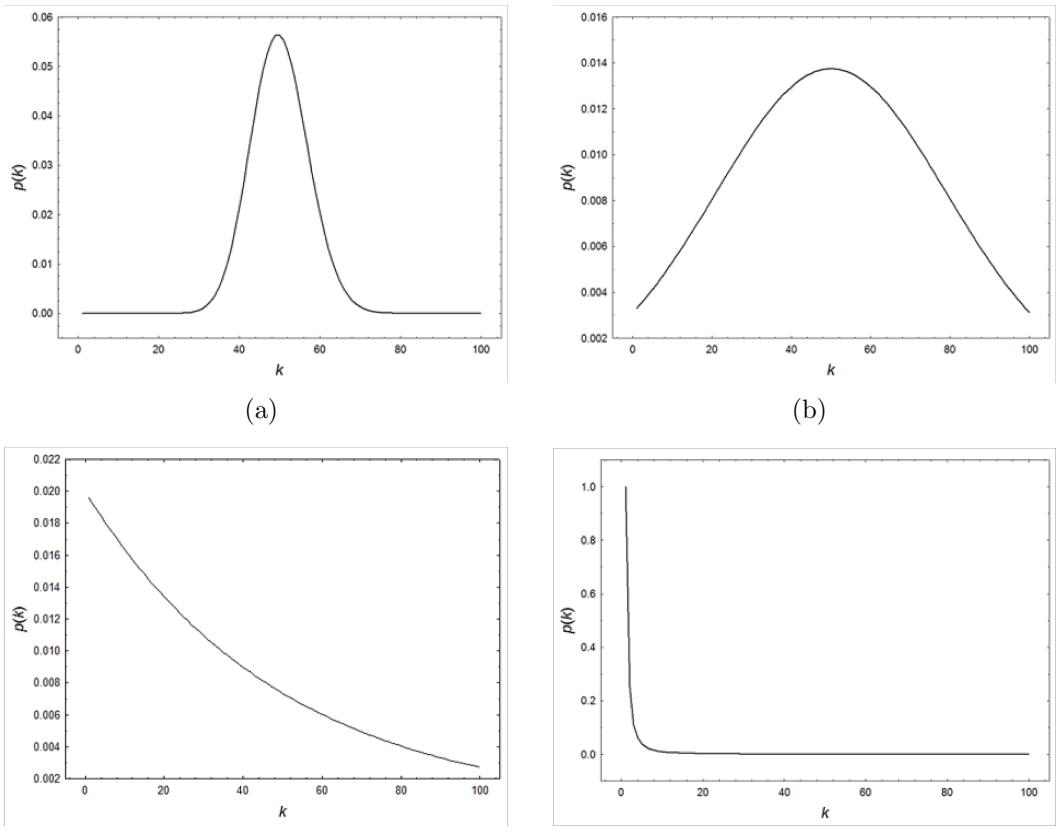
quantity in the study of complex networks, this is called the *average node degree* and is defined as $\bar{k} = \frac{1}{n}\mathbf{1}^T\mathbf{k} = \frac{1}{n}\sum_{i=1}^{n} k_i$. When the degrees are arranged in a non-increasing order, then this array is referred to as the *degree sequence* of the network. One useful way of representing the information about node degrees is with the diagonal matrix $\mathbf{K}$ known as the *degree matrix* of the network which contains the degree of the nodes in the main diagonal.

The distribution of the node degrees provides some properties of the structure of a network. If we represent with $p(k)$ the probability that a node, chosen uniformly by random, has degree k in a network, then this quantity is equal to $p(k) = \frac{n(k)}{n}$ where $n(k)$ is the number of nodes having degree k and n is the total number of nodes (or *size*) of the network. A plot of $p(k)\,\text{vs}\,k$ gives us the form of the degree distribution of the network, and if we plot this relation in a log-log scale we can have an idea about the kind of statistical distribution that these node degrees follow. There are some degree distributions commonly found in the study of complex networks which are shown in figure 1.1.6. Nonetheless, this information should be taken with caution because sometimes the information available to study the degree distribution of a network, i.e. the number of points to depict the plot $p(k)\,\text{vs}\,k$, is limited due the variability of node degrees in general is not so high compared to the size of the network, this makes difficult to accurately fit the data to a well known probability distribution.

There is one degree distribution that has attracted the attention of the scientific community: the power-law degree distribution. The characteristic of this distribution is that the probability of finding a node with degree k decays as a negative power of the degree (see fig. 1.1.6). Its expression is $p(k) \sim k^{-\gamma}$ which means that the probability of finding a high-degree nodes is relatively small compared with the high probability of finding low-degree nodes. The networks that show this form of degree distribution are called "scale-free" networks referring to the existence of a power-law relation $p(k) = Ak^{-\gamma}$ between the probability and the node degree which when scaling the degree by a constant factor c results in a proportionate scaling of the function $p(k,\, c) = A(ck)^{-\gamma} = Ac^{-\gamma} \cdot p(k)$.

As with all degree distributions, power-law relations can be represented on a logarithmic scale, this representation resembles a straight line of the form $\ln p(k) = -\gamma \ln k + \ln A$ where $-\gamma$ is the slope and $\ln A$ is the intercept of the function.

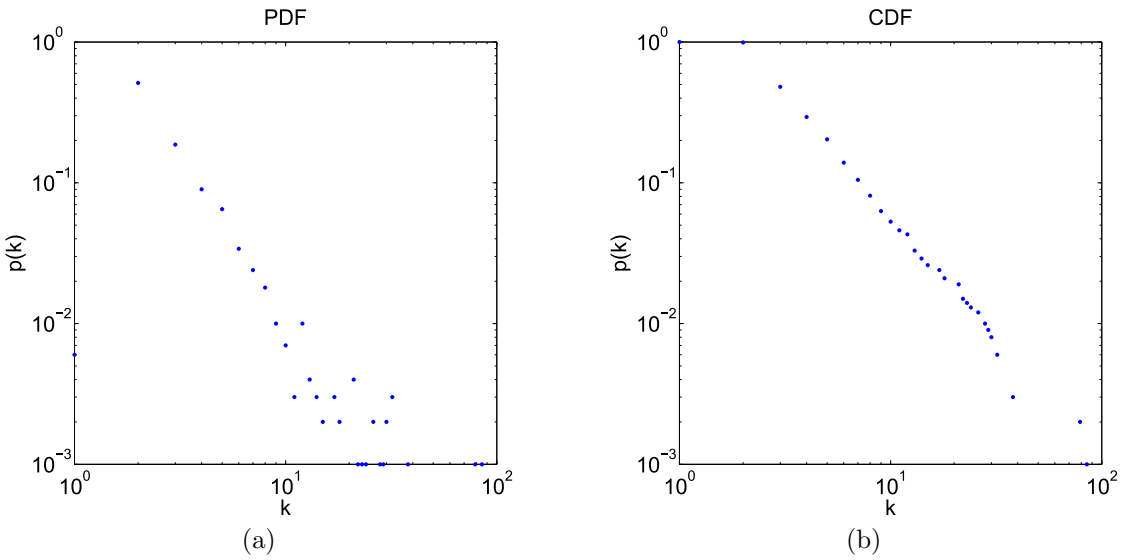
Frequently when plotting $p(k)$ vs k in a log-log scale (see fig. 1.1.7) the tail of the distribution is very noisy. One approach to reduce this noise is to use instead the cumulative distribution function CDF defined as $P(k) = \sum_{k'=k}^{\infty} p(k')$, which represents the probability of choosing at random a node with degree greater than, or equal to k. For the case of power-law degree distributions $P(k)$ also exhibits a power-law decay of the form $P(k) \sim \sum_{k'=k}^{\infty} k'^{-\gamma} \sim k^{-(\gamma-1)}$, this means that the log-log plot of $P(k)$ vs k will be a straight line too for the scale-free networks.



(a) (b)

(c) (d)

Figure 1.1.6: Some common degree distributions in networks. (a) Poisson distribution (b) Gaussian distribution (c) Exponential distribution (d) Power-law distribution



(a) (b)

Figure 1.1.7: (a) Probability distribution function and (b) cumulative distribution function of a network with power-law degree distribution

Degree-degree correlations and assortativity

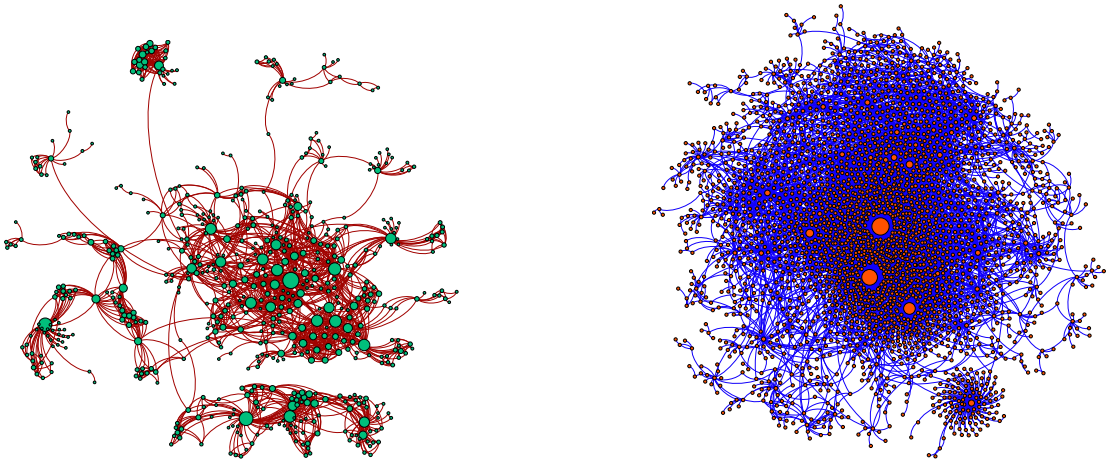
There exist different kinds of degree-degree correlations in complex networks, these correlations give information about the way nodes with a certain degree are connected in a network. One approach to measure this characteristic was proposed in [33] by Newman and it was called assortativity coefficient which is based on the Pearson coefficient of the degree-degree correlation, this means that the assortativity coefficient calculates the correlation the correlation coefficient for the degrees of the existing nodes at both sides of every link in a network. Formally, if we let $e(k_i k_j)$ be the faction of links connecting a node of degree k_i to a node of degree k_j, the probability that a node selected by random in the network has degree k_j is equal to $p(k_j)$. For mathematical convenience, in formulating the concept, Newman considered the degree minus one instead of the degree of the nodes, and called this "excess degrees". Thus, the distribution of the excess degree of a node at the end of link chosen by random is

$$q(k_j) = \frac{(k_j + 1)\, p(k_j + 1)}{\sum_i k_i\, p(k_i)}. \tag{1.1.6}$$

Then the assortative coefficient is defined as

$$r = \frac{\sum_{k_i k_j} k_i k_j [e(k_i k_j) - q(k_i) q(k_j)]}{\sigma_q^2} \tag{1.1.7}$$

where σ_q^2 is the standard deviation of the distribution $q(k_j)$. This coefficient is undefined for the case when the standard deviation is equal to zero, this is when all nodes have the same degree, i.e., the case of regular networks. When positive values are obtained from the coefficient it is said that the network displays assortative mixing of degrees and when the values obtained are negative, we say that the network displays dissasortative mixing of degrees. The first case, assortative mixing, means that low-degree nodes are preferentially attached to other low-degree nodes and high-degree nodes are preferentially joined to other high-degree nodes (see fig. 1.1.8a). For the second case, dissasortative mixing, those nodes with high degree are preferentially linked to low-degree nodes (see fig. 1.1.8b). This degree mixing characteristic has implications for network resilience to intentional attacks, epidemic spreading, synchronisation, cooperation and diffusion (see [5] and references therein).



(a) (b)

Figure 1.1.8: Example of networks with assortative and dissasortative mixing of degrees. The radii of the nodes have been drawn proportionally to their degrees. (a) Drugs users network shows assortative mixing (b) Component of the Internet in 1997 shows dissasortative mixing

Shortest path distance

For an undirected network, the shortest path distance $d(u, v)$ is the number of links in the shortest path between the nodes u and v. When those nodes u and v belong to different connected components, the distance between them is set to infinite, this is $d(u, v) \coloneqq \infty$. The values of distance among all nodes in a network can be presented in matrix form, the result, for an undirected network, is a square symmetric matrix known as the distance matrix $\boldsymbol{D}$, and the distances between any node v and the rest of them are given at the vth row (or column) of the matrix. For the case of directed networks, $\mathbf{D}$ is not necessarily symmetric and it may contain entries equal to infinite. The *eccentricity* of a node in a network is the maximum value in the correspondent row (or column) of the distance matrix, this is for the node v [5]:

$$e(v) = \max_{x \in V(G)} \{d(v,\, x)\}\,. \tag{1.1.8}$$

The maximum eccentricity among the nodes of a network is known as the *diameter* of the network, and it is defined as

$$diam(G) = \max_{x,y \in V(G)} \{d(x,\, y)\}\,. \tag{1.1.9}$$

The radius of the network is the minimum eccentricity of the nodes. A node is called central if its eccentricity is equal to the radius of the network, and the centre of the graph is constituted by the set of all central nodes [5].

The sum of all entries of a row (column) of the distance matrix $\boldsymbol{D}$ is referred to as the *distance sum* of the correspondent node, and is defined as

$$s(u) = \sum_{v \in V(G)} d(u,\, v) \tag{1.1.10}$$

this quantity is also know as the total distance or *status* of the node.

Average path length

A common metric related to the shortest paths from a vertex u to any other vertex v in $G = (V, E)$ is the average distance between vertices referred to as the average path length $\bar{d}$ defined as

$$\bar{d}(G) = \frac{1}{n(n-1)} \sum_{u,v \in V, u \neq v} d(u, v) \tag{1.1.11}$$

where the average is taken considering only those pairs of nodes for which there is a path connecting them. The average path length is bounded as $1 \leq \bar{d}(G) \leq \frac{n+1}{3}$, where the lower bound is reached for the complete network and the upper bound happens for the path with n nodes [5].

Clustering coefficient

A metric that is commonly used in complex networks analysis is the clustering coefficient, this concept was originally proposed by Watts and Strogatz in 1998 [34]. The idea behind this metric is: in a network $G = (V, E)$ we want to see for a given vertex $v \in V$, to what extent the neighbours of v are also neighbours of each other, this is to what extent the vertices adjacent to v as also adjacent to each other [35].

Formally, for a given node i, the clustering coefficient is the number of triangles connected to this node $|C_3(i)|$ divided by the number of triples centred on it: [5]

$$C_i = \frac{2\,|C_3(i)|}{k_i(k_i - 1)} \tag{1.1.12}$$

where k_i is the degree of the node i. The average value of the clustering for all nodes n in a network, commonly referred to as the average clustering coefficient, is then defined as

$$\bar{C} = \frac{1}{n} \sum_{i=1}^{n} C_i. \tag{1.1.13}$$

There is an alternative coefficient defined by Newman in 2001 [36], this globally characterises the cliquishness of a network by considering that each triangle consists of three different connected triples, one with each of the nodes at the centre [5]. This index is known as network transitivity and is defined as

$$C = \frac{3\,|C_3|}{|P_2|} \tag{1.1.14}$$

this is, the ratio of three times the number of triangles C_3 divided by the number of connected triples P_2 (2-paths).

It is worth to point out that a network with large average clustering ($\bar{C}$) is not necessarily a high clustered network if measured by C, this is, a network can be highly clustered at a local level but not on a global scale [5].

1.2 Matrices and networks

Matrices are useful constructs when dealing with large arrays of numbers, and are a key tool for the study of complex networks. There are some matrices that frequently appear in networks and give important structural information: the adjacency and the Laplacian matrices, in this section we define both matrices as they play a key role in our work.

1.2.1 Adjacency matrix

The adjacency information of the nodes of a network $G = (V, E)$ can be represented by the adjacency matrix $\boldsymbol{A}$ defined as

$$A_{ij} = \begin{cases} 1 & if\ i, j \in E \\ 0 & otherwise \end{cases} \tag{1.2.1}$$

the dimension of this matrix is $|V| \times |V|$. For an undirected network, the vth row (column) of the adjacency matrix has exactly k_v entries which corresponds to the

number of links or nearest neighbours the node v has, this is the degree of the node. The column vector of node degrees $\boldsymbol{k}$ can be obtained from the matrix $\boldsymbol{A}$ and a $|V| \times 1$ all-one vector $\mathbf{1}$ as follows:

$$\boldsymbol{k} = (\mathbf{1}^T \boldsymbol{A})^T = \mathbf{A}\mathbf{1}. \tag{1.2.2}$$

For a directed network, the adjacency matrix could be not symmetric as the element A_{ij} is equal to one if there is a link pointing from i to j, this characteristic implies that the product $\mathbf{1}^T \boldsymbol{A}$ is not necessarily equal to $\mathbf{A}\mathbf{1}$, which lead us to have two distinct types of node degrees: the *in-degree* and the *out-degree*. For a node i the first node degree corresponds to the number of links that are pointing to i, and the second one corresponds to the number of links coming from i. Both node degrees can be obtained in vector form as

$$\boldsymbol{k}^{\mathbf{in}} = (\mathbf{1}^T \boldsymbol{A})^T \tag{1.2.3}$$

$$\boldsymbol{k}^{\boldsymbol{out}} = \mathbf{A}\mathbf{1}. \tag{1.2.4}$$

When the network is weighted, the entry A_{ij} of $\boldsymbol{A}$ corresponds to the weight w_{ij} associated to the link between nodes i and j and for this case, the weighted in- and out- degrees can be computed following the expressions (1.2.3) and (1.2.4) with $\boldsymbol{A}$ being the weighted adjacency matrix of the network.

There is another matrix which has a close relation with the adjacency matrix, this is called the incidence matrix $\boldsymbol{B}$ and has elements $B_{v_i e} = 1$ if the node v_i is incident with the link e, the dimension of $\boldsymbol{B}$ is $|V| \times |E|$. Then, the adjacency matrix $\boldsymbol{A}$ is related with the incidence matrix $\boldsymbol{B}$ and the degree matrix $\mathbf{K}$ (defined on page 20) in the following way:

$$\mathbf{A} = \boldsymbol{B}\boldsymbol{B}^T - \boldsymbol{K}. \tag{1.2.5}$$

The adjacency and degree matrices of a network are related with the average node

degree in a lemma, known as the *handshaking lemma*, which states that the sum of all node degrees is equal to twice the number of links [5], this is

$$\mathbf{1}^T \boldsymbol{A}\mathbf{1} = \boldsymbol{K}^T\mathbf{1} = \sum_{i=1}^{n} k_i = 2\,|E| \tag{1.2.6}$$

thus, the average node degree is

$$\bar{k} = \frac{2\,|E|}{n}. \tag{1.2.7}$$

1.2.2 Laplacian matrix

The first recognizable appearance of the Laplacian matrix may be in what has come to be know as *Kirchhoff's matrix tree theorem* and it has been referred to as *Kirchhoff matrix*, *matrix of admittance*, *information matrix*, *Zimm matrix*, *Rouse-Zimm matrix*, *connectivity matrix* and *vertex-vertex incidence matrix*, among other names depending on the application field [37]. But perhaps the origins of the justification for the name Laplacian comes from a study of vibrations of membranes related with the question whether one could "hear the shape of a drum" posed by Mark Kac in 1966 [37, 38].

This matrix is important for the study of networks because of the application of its spectral properties to tackle problems like clustering, pattern recognition, consensus algorithms, synchronisation, information theory, expanders, among others [39].

The concept of the continuous Laplacian operator $\Delta = \nabla^2$ can be extended to the field of networks trough the definition of a derivative as the limit of differences [40]

$$f'(a) = \lim_{x \to a} \frac{f(x) - f(a)}{x - a}. \tag{1.2.8}$$

This extension of the Laplacian operator to the discrete case of networks is known as the *graph Laplacian*.

1.2.2.1 Definition

Let f be a function defined at the node v of a network $G = (V, E)$ with arbitrary orientation of every link. We can represent the network trough an oriented incidence matrix $\nabla(G)$ defined as

$$\nabla_{ij} = \begin{cases} +1 & if\, node\, v_i\, is\, the\, positive\, end\, of\, link\, e_j \\ -1 & if\, node\, v_i\, is\, the\, negative\, end\, of\, link\, e_j \\ 0 & otherwise \end{cases}. \tag{1.2.9}$$

This matrix represents the gradient operator for the network which gives the maximum rate of change of the function f. Then if we let $f : V \to \mathbb{R}$ be an arbitrary function, then $\nabla f : E \to \mathbb{R}$ is given by [5]

$$(\nabla f)(e) = f(u) - f(v) \tag{1.2.10}$$

where u and v are, correspondingly, the starting and ending points of the oriented link e.

We recall that a vector field is a function on the interval with an orientation. For this case, the interval corresponds to the link in the network, which together with its orientation forms a vector field. Thus, the continuous analogous of $\nabla(G)$ is the gradient $(\nabla f) = \frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}, \ldots, \frac{\partial f}{\partial x_n}$, which gives the maximum rate of change of the function with direction.

We now consider the operator $Lf = -\nabla \cdot (\nabla f)$, which is the nertwork version of the Laplacian operator

$$\nabla^2 f = \frac{\partial^2 f}{\partial x_1^2} + \frac{\partial^2 f}{\partial x_2^2} + \cdots + \frac{\partial^2 f}{\partial x_n^2}. \tag{1.2.11}$$

Then the Laplacian operator acting on $L : \mathbb{R}^{|V|} \to \mathbb{R}^{|V|}$ is defined as

$$(Lf)(u) = \sum_{u \sim v} [f(u) - f(v)], \tag{1.2.12}$$

which in matrix form is given by

$$L_{uv} = \sum_{e \in E} \nabla_{eu} \nabla_{ev} = \begin{cases} -1 & if\ uv \in E \\ k_u & if\ u = v \\ 0 & otherwise \end{cases} . \tag{1.2.13}$$

The graph Laplacian can also be written in terms of the adjacency matrix by taking into account (1.2.13) as

$$\mathbf{L} = \boldsymbol{K} - \boldsymbol{A} \tag{1.2.14}$$

being $\boldsymbol{A}$ the adjacency matrix of the network G and $\boldsymbol{K}$ the degree matrix of the same network.

1.2.2.2 Spectra and properties of the Laplacian matrix

The tools of algebraic graph theory associate algebraic objects such as matrices and polynomials, to graphs, and provide algebraic techniques for their study. From this association, it has been constituted the study of spectral properties of matrices representing graphs that we know as *spectral graph theory* [3]. This branch of graph theory focuses on the study of the eigenvalues associated with the matrix representation of graphs. Regarding the Laplacian matrix, its spectrum and eigenvalues can reveal important properties for dynamical problems on networks [39].

From (1.2.5) we have that $\boldsymbol{L} = \boldsymbol{K} - \boldsymbol{A} = \mathbf{B}\mathbf{B}^T$. This matrix is symmetric and positive semidefinite, this makes its spectrum real, which is defined as the set of its eigenvalues together with their multiplicities [40]. We can order its eigenvalues as

$$\mu_1(\mathbf{L}) \leq \mu_2(\mathbf{L}) \leq \cdots \leq \mu_n(\mathbf{L}) \tag{1.2.15}$$

with $\mu_1(\boldsymbol{L}) = 0$,

then its spectrum can be written as

$$Sp\boldsymbol{L} = \begin{pmatrix} \mu_1(\boldsymbol{L}) & \mu_2(\boldsymbol{L}) & \cdots & \mu_n(\boldsymbol{L}) \\ m(\mu_1(\boldsymbol{L})) & m(\mu_2(\boldsymbol{L})) & \cdots & m(\mu_n(\boldsymbol{L})) \end{pmatrix} \tag{1.2.16}$$

where $m(\mu_1(\boldsymbol{L}))$, $m(\mu_2(\boldsymbol{L})), \cdots, m(\mu_n(\boldsymbol{L}))$ are the multiplicities of the correspondent eigenvalues.

The eigenvalues of $\boldsymbol{L}$ are bounded as

$$0 \leq \mu_j(\boldsymbol{L}) \leq 2k_{max} \tag{1.2.17}$$

and

$$\mu_n(\boldsymbol{L}) \geq k_{max} \tag{1.2.18}$$

being k_{max} the maximum node degree of the network.

The multiplicity of 0 as an eigenvalue of $\boldsymbol{L}$ is equal to the number of connected components in a network and the second smallest eigenvalue, $\mu_2(\boldsymbol{L})$, is commonly referred to as the algebraic connectivity of the network [41].

Theorem 1.1. *The graph G is connected if and only if $\mu_2(\boldsymbol{L}) > 0$.*

Proof. Since the null spaces of $\mathbf{B}^T$ and $\mathbf{L}$ are the same, it suffices to show that the null space of $\mathbf{B}^T$ has dimension one when the graph $G = (V, E)$ is connected. Suppose that there exists a vector $\mathbf{z} \notin \mathbf{span}\{\mathbf{1}\}$, with $\mathbf{1}$ being the vector with ones in all its entries, such that

$$\mathbf{z}^T\mathbf{B} = 0,$$

that is, when $uv \in E$ then $z_v - z_u = 0$. However, since G is connected, this implies that $z_v = z_u$ for all $u, v \in V$ and $\mathbf{z} \in \mathbf{span}\{\mathbf{1}\}$. Thus, the dimension of the null space of $\mathbf{B}^T$ is one if and only if the geometric, and hence algebraic, multiplicity of the zero eigenvalue of the Laplacian, namely $\mu_1(\mathbf{L})$, is one [3]. □

Some expressions for the spectra of different kinds of simple graphs are: [5]

Path P_n : $\mu_j(\mathbf{L}) = 2 - 2\cos\left(\frac{\pi(j-1)}{n}\right)$, $i = 1, \ldots, n$.

Cycle C_n : $\mu_j(\mathbf{L}) = 2 - 2\cos\left(\frac{2\pi j}{n}\right)$.

Star S_n : $Sp(\mathbf{L}) = \{0\, n^{n-2}\, n\}$.

Complete K_n : $Sp(\mathbf{L}) = \{0\, n^{n-1}\}$.

Complete (bipartite) $K_{n1,n2}$: $Sp(\mathbf{L}) = \left\{0\, n_1^{n_2-1}\, n_2^{n_1-1}\right\}$.

Analysing the eigenvectors associated to the eigenvalues of $\boldsymbol{L}$ is also part of spectral graph theory. If we let Λ be a diagonal matrix of eigenvalues of the Laplacian matrix ordered as in the expression(1.2.15), then [5]

$$\Lambda = diag\left(\mu_1(\boldsymbol{L}),\, \mu_2(\boldsymbol{L}), \ldots,\, \mu_n(\boldsymbol{L})\right) \tag{1.2.19}$$

and let $\boldsymbol{\Phi}$ be a matrix whose columns are orthonormal eigenvectors $\boldsymbol{\phi}_1,\, \boldsymbol{\phi}_2, \ldots,\, \boldsymbol{\phi}_n$. Then the spectral decomposition of the Laplacian matrix is give by:

$$\boldsymbol{L} = \boldsymbol{\Phi}\Lambda\boldsymbol{\Phi}^T. \tag{1.2.20}$$

The eigenvector of $\boldsymbol{L}$ associated with the 0 eigenvalue in a connected network is $\boldsymbol{\phi}_1(\boldsymbol{L}) = \frac{1}{\sqrt{n}}\mathbf{1}$, and if $0 \neq \mu < n$ is an eigenvalue of the Laplacian, then the eigenvector associated with μ takes the value 0 on every node of degree $n - 1$ [5].

The eigenvectors of the Laplacian contain useful information related to the structure of a network. The spectrum of this important matrix leads to the concept of algebraic connectivity of a network which is assigned to the eigenvalue μ_{n-1}, and the eigenvector associated with this value has an important property.

Theorem 1.2. (Fiedler's theorem of connectivity of spectral graph partitions) *Suppose $G = (V, E)$ is a connected network with graph Laplacian $\mathbf{L}$ whose second smallest eigenvalue is $\mu_{n-1} > 0$. Let x be the eigenvector associated with μ_{n-1}. Let $r \in \mathbb{R}$ to partition the nodes in V into two sets*

$$V_1 = \{\, i \in V \mid x_i \geq r \,\},\ V_2 = \{\, i \in V \mid x_i < r \,\}.$$

Then the subgraphs of G induced by the sets V_1 and V_2 are connected.

This result provides a method for partitioning a network in a symmetric way assuring that such partitions are still connected. One form of choosing r is to be the median value of x, this guarantees that the clusters are evenly sized. The vector x is referred to as the *Fiedler vector*.

Dynamics on Networks

The concept of dynamical system is rather general, it refers to anything which evolves with time. A communication network is a dynamical system, vehicles such as aircraft, spacecraft, motorcycles, cars, among others, are dynamical systems. Other examples include machines, robots, chemical plants, electrical circuits, even structures like bridges (think of a structure subject to strong winds or an earthquake). But the concept of dynamical system is not restricted to engineering systems, it can be applied to plants, group of animals, human beings or the economy of one or more countries. The interactions among the parts that constitute a network are dynamical processes, which can be considered in a continuous or discrete time basis. Here we present a brief overview of dynamical systems (complete information can be consulted from the references given), and two examples of how these kind of processes can be represented on networks: epidemic spreading and synchronisation. We finish this part by setting the theoretical grounds of the main model used in this thesis, the consensus protocol.

1.3 Dynamical Systems

A dynamical system can be defined as a triple $\mathbb{S} = (\mathbb{T}, \mathbb{W}, \mathfrak{B})$, where $\mathbb{T}$ is a subset of $\mathbb{R}$, called the time axis, $\mathbb{W}$ is a set called the signal space, and $\mathfrak{B}$ is a subset of $\mathbb{W}^{\mathbb{T}}$ called the behaviour[1] [42].

The set $\mathbb{T}$ specifies the set of times instances relevant to our problem. Usually $\mathbb{T}$ equals $\mathbb{R}$ or $\mathbb{R}_+$ in continuous-time systems, and $\mathbb{Z}$ or $\mathbb{Z}_+$ in discrete-time systems. The set $\mathbb{W}$ specifies the way in which the outcomes of the signals produced by the dynamical system are formalised as elements of a set, this is, these outcomes are the variables whose evolution in time we are describing [42]. The behaviour $\mathfrak{B}$ is a family of time trajectories taking their values in the signal space, in other words, the elements of $\mathfrak{B}$ are the trajectories compatible with the laws that govern the system. In most models the behaviour is described by equations, and then this set is formed by those elements satisfying a set of equations which often take the form of differential equations for continuous-time models, and difference equations for discrete-time models.

[1] $\mathbb{W}^{\mathbb{T}}$ is standard mathematical notation for the collection of maps from $\mathbb{T}$ to $\mathbb{W}$ [42]

Then a dynamical system can be thought of as any system whose state, represented by some set of quantitative variables, changes over time according to some given rules or equations [4]. A simple example of a continuous dynamical system described by a single real variable $x(t)$ that evolves according to a first-order differential equation is

$$\frac{dx}{dt} = \dot{x} = f(x), \tag{1.3.1}$$

where $f(x)$ is some specified function of x. Commonly, there is an initial condition that specifies the value of variable x at some initial time t_0, and is expressed as x_0.

A system interacts with its environment through inputs and outputs. Inputs can be considered to be exerted on the system by the environment, whereas outputs are exerted by the system on the environment. A fundamental concept for describing the behaviour of a dynamical system is the state of the system.

When casting a situation into a mathematical expression, we need to identify the salient features of the situation and attaching mathematical objects to them, this means, we try to associate a number or collection of numbers with the situation which characterises it [43]. For convenience, we think of these numbers as vectors in $\mathbb{R}^n$ which can contain n numbers associated to properties of interest from the situation under analysis. In this case, each conceivable situation corresponds to a point in $\mathbb{R}^n$ (or some subset of it). We called the set of all points in $\mathbb{R}^n$ corresponding to a conceivable situation the *state space*.

As change is an inherent feature of our world and the situations we encounter, in general the state space description of an input-output system usually involves a time variable t and three sets of variables:

- State variables: $x_{1,}x_{2,}\ldots,\ x_n$
- Input variables: $u_1,\ u_2,\ldots,\ u_m$
- Output variables: $y_1,\ y_2,\ldots,\ y_p$

The basic representation for linear systems is the linear state-equation written in the standard form as

$$\begin{aligned} \dot{x}(t) &= A(t)x(t) + B(t)u(t) \\ y(t) &= C(t)x(t) + D(t)u(t) \end{aligned} \quad . \tag{1.3.2}$$

The $n \times 1$ vector function of time $x(t)$ is called the *state vector*, and its components, $x_1(t), \ldots, x_n(t)$, are the state variables. The input signal is the $m \times 1$ function $u(t)$, and $y(t)$ is the $p \times 1$ output signal, assuming that p, $m \leq n$ [44].

The default assumption on the coefficients matrices in (1.3.2) is that their entries are continuous, real-valued functions defined for all $t \in [0, \infty)$, thus the expression (1.3.2) is referred to as linear time varying system. In a continuous-time system, the time-variable t can be any real number, and for a discrete-time system, the time-variable only takes integer values.

For the case when the entries of the coefficients matrices are constant, the expression (1.3.2) is expressed as

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) + Du(t) \end{aligned} \tag{1.3.3}$$

which is called linear time invariant system, and in scalar terms is described by:

$$\begin{aligned} \dot{x}_1 &= a_{11}x_1 + \cdots + a_{1n}x_n + b_{11}u_1 + \cdots + b_{1m}u_m \\ &\vdots \\ \dot{x}_n &= a_{n1}x_1 + \cdots + a_{nn}x_n + b_{n1}u_1 + \cdots + b_{nm}u_m \end{aligned} \tag{1.3.4}$$

and

$$\begin{aligned} y_1 &= c_{11}x_1 + \cdots + c_{1n}x_n + d_{11}u_1 + \cdots + d_{1m}u_m \\ &\vdots \\ y_p &= c_{p1}x_1 + \cdots + c_{pn}x_n + d_{p1}u_1 + \cdots + d_{pm}u_m \end{aligned} \quad . \tag{1.3.5}$$

The group of equations in (1.3.4) are referred to as the *state-equations*, and the group (1.3.5) are referred to as the *output-equations*.

A system with no inputs is described by

$$\dot{x} = Ax \tag{1.3.6}$$

which is called autonomous because the right-hand side of the equations do not depend explicitly on time t.

1.3.1 Discrete dynamical systems

Since our work deals with a discrete model of consensus, we briefly review some concepts of discrete dynamical systems. The idea of being *discrete* is directly related with the concept of *discrete time,* this is, we suppose that we measure time in discrete units, these units might be hours, seconds, tenths of a second, years, centuries or any other period of time which, once fixed, we only allow ourselves to consider integer multiples of them.

We represent the state of a system at time $t = 0$ by $x(0)$ and the state of the system at the k-th time step as $x(k)$. A discrete dynamical system is a system in which the state at any time depends only on the state a the time before [43], then a discrete dynamical system can be expressed as

$$x(k+1) = f\left(x(k)\right), \tag{1.3.7}$$

where f is a function from $\mathbb{R}^n$ to $\mathbb{R}^n$ non explicitly dependent on k.

Discrete dynamical systems arise from continuous dynamical systems by assuming that rates of change are constant over some small time interval, so the moment we choose at Δt, we transform a continuous time dynamical system $\dot{x} = f(x)$ (1.3.1) into the discrete time dynamical system

$$x(k+1) = x(k) + (\Delta t)\, f\left(x(k)\right) \tag{1.3.8}$$

with time unit equal to Δt. It is worth to note that this expression means, in a more simplistic way, that the *new* or *future* value of the variable x depends on its *old* or *past* value:

$$x_{new} = x_{old} + (\Delta t)\, x'_{old}. \tag{1.3.9}$$

One way of converting a continuous time dynamical model to a discrete time form is by applying a finite difference approximation [45]. We describe the general process for obtaining the discrete time version of a dynamical system as this is the form we use for our performing our simulations.

Consider a nonlinear differential equation

$$\frac{dx(t)}{dt} = f(x, u) \tag{1.3.10}$$

where x is the output variable and u is the input variable. The previous equitation can be numerically integrated by introducing a finite difference approximation for the derivative. For our purposes, lets consider the first order difference approximation to the derivative at $t = k\Delta t$, this is

$$\frac{dx(t)}{dt} \approxeq \frac{x(k) - x(k-1)}{\Delta t} \tag{1.3.11}$$

were Δt is the integration interval (specified by the user) and $x(k)$ denotes the value of $x(k)$ at $t = k\Delta t$. Substituting equation 1.3.10 into equation 1.3.11, and evaluating $f(x, u)$ at the previous values of x and u, i.e., $x(k-1)$ and $u(k-1)$, gives

$$\frac{x(k) - x(k-1)}{\Delta t} \approxeq f(x(k-1), u(k-1)) \tag{1.3.12}$$

or, in other form

$$x(k+1) = x(k) + (\Delta t) f(x(k)),\ u(k) \tag{1.3.13}$$

which is the discrete time form of equation 1.3.8 used for the simulations in this thesis.

A solution for a discrete time dynamical system is a sequence $(x(0), x(1), \cdots)$ which satisfies the expression $x(k+1) = f(x(k))$. A fixed state vector referred to as x^e is called an equilibrium state, which is constant for all k, this means that

$$x(k) \equiv x^e. \tag{1.3.14}$$

The equivalent version of (1.3.3) for the discrete time case is

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) \ , \\ y(k) &= Cx(k) + Du(k) \end{aligned} \tag{1.3.15}$$

where the n-vector $x(k)$ is the state vector at time k, the m-vector $u(k)$ is the input vector at time k, and the p-vector $y(k)$ is the output vector at time k. The matrix A is generally referred to as the system matrix and its dimension is $n \times n$. The matrices B, C and D have, respectively, dimensions $n \times m$, $p \times n$ and $p \times m$. This means that in scalar terms we have

$$\begin{aligned} x_1(k+1) &= a_{11}x_1(k) + \cdots + a_{1n}x_n(k) + b_{11}u_1(k) + \cdots + b_{1m}u_m(k) \\ &\vdots \\ x_n(k+1) &= a_{n1}x_1(k) + \cdots + a_{nn}x_n(k) + b_{n1}u_1(k) + \cdots + b_{nm}u_m(k) \end{aligned} \tag{1.3.16}$$

and

$$\begin{aligned} y_1 &= c_{11}x_1 + \cdots + c_{1n}x_n + d_{11}u_1 + \cdots + d_{1m}u_m \\ &\vdots \\ y_p &= c_{p1}x_1 + \cdots + c_{pn}x_n + d_{p1}u_1 + \cdots + d_{pm}u_m \end{aligned} \ . \tag{1.3.17}$$

Thus, a system with no inputs is described by

$$x(k+1) = Ax(k). \tag{1.3.18}$$

1.4 Dynamical processes on networks

Many real world phenomena can be represented as dynamical systems on networks, for example, the spread of information among a group of people, the movement of money in a country's economy, the traffic on roads, the flow of electricity over a grid, the evolution of population in a ecosystem, among other situations [4].

Two main approaches are adopted when dealing with dynamical processes on networks [46]. In the first approach, we can identify each node of the network with a single individual or element of the system, and in the second approach, we consider dynamical entities such as people, information packets, energy or matter flowing through a network whose nodes identify locations where the dynamical entities transit. In both approaches, the dynamical description of the system can be achieved by introducing for each node i a corresponding variable x_i that characterises its dynamical state.

Then, if each node represents a single individual, the variable x_i may describe a particular attribute of the individual. For the case of dynamical entities moving in a network, the state variable x_i generally depends on the entities present at that node. We can enumerate all possible states $x_i = 1, 2, \ldots, \kappa$ for each node, and the knowledge of the state variable of all nodes in the network defines the (microscopic) state of the system [46].

Thus, we can denote a particular configuration of the network at time t by the set $x(t) = (x_1(t), x_2(t), \ldots, x_n(t))$, where n is the number of nodes in the network.

It is common that when referring to dynamics on networks we consider to have independent dynamical variables on each node and that they are coupled together along the edges of the network, this is, when writing an equation for the time evolution of a variable x_i the individual terms appearing in that equation involve x_i, other variables on vertex i, or one or more variables on a vertex adjacent to i in the network [4].

In real systems, adding the dimension of dynamics to the characterisation of networks allows a better comprehension of the systems under analysis through considering the interplay between structural and dynamical aspects [47]. We briefly

describe the general reasoning existing under a couple of well known examples of dynamics on networks: the epidemic spreading and synchronisation models.

1.4.1 Epidemic spreading

Epidemic spreading has to do with the modelling for the spread of a particular infectious disease in a population, with the objective of reproducing the actual dynamics of the disease, thus being able to design proper strategies for controlling and possibly eradicating the infection. One of the first works studying the effects of the network topology on the patterns of a disease spread was form Pastor-Satorras and Vespignani [48] where they analysed the effects of the topology of the internet on the spreading of viruses. For disease spreading, there are two models that commonly arise when studying this topic: the susceptible-infected-removed (SIR) and the susceptible-infected-susceptible (SIS) models. We briefly describe the ideas behind these models.

The approach for modelling epidemic spreading is based on compartmental models, which are those models in which the individuals in a population are divided into a set of different groups. The SIR model describes the phenomenon of diseases as a result of the immunisation or death of infected individuals, assuming that each of them can be in one out of three possible sates:

1. Susceptible ($\mathfrak{S}$): these are healthy people that can catch the disease if exposed to infected persons.
2. Infected ($\mathfrak{I}$): those individuals that already have the disease.
3. Removed ($\mathfrak{R}$): this term referrers to individuals that cannot get (or passes it on) the disease, either because she becomes immune or dies.

Once an individual catches the infection, she moves into the infected class. This model is based on two parameters: the transmission rate $\mathfrak{t}$ and the recovery rate $\mathfrak{r}$, and it assumes that the individuals live at the sites of a given network. Then the process starts via infection of neighbouring individuals, which happens with a rate $\mathfrak{t}$ and is expressed as

$$\mathfrak{S}(i) + \mathfrak{I}(j) \xrightarrow{\mathfrak{t}} \mathfrak{I}(i) + \mathfrak{I}(j), \tag{1.4.1}$$

where i and j are neighbours, and the recovery occurs with a rate $\mathfrak{r}$ and is expressed as

$$\mathfrak{I}(i) \xrightarrow{\mathfrak{r}} \mathfrak{R}(i). \tag{1.4.2}$$

Since in reality not all diseases confer immunity to their survivors, an individual can catch the same disease more than once, this situation is reproduced by the SIS model by considering only two states in the process: susceptible ($\mathfrak{S}$) and infected ($\mathfrak{I}$). In the SIS model, susceptible individuals catch the infection and move into the infected state, becoming again susceptible after a period of time in which they recover, and finally, they are again exposed to the epidemic. The expressions for the SIS model are

$$\mathfrak{S}(i) + \mathfrak{I}(j) \xrightarrow{\mathfrak{t}} \mathfrak{I}(i) + \mathfrak{I}(j) \tag{1.4.3}$$

which reflects the infectious part of the process, and

$$\mathfrak{I}(i) \xrightarrow{\mathfrak{r}} \mathfrak{S}(i) \tag{1.4.4}$$

that accounts for the consideration that an infected individual becomes susceptible after a period of time.

These two models have different dynamics once an infection is introduced into a population. For the SIR model, the maintenance of the infection in a closed population is impossible due to the depletion of susceptible as the epidemic spread through the population, meanwhile in the SIS model the disease can persist indefinitely circulating around the population. As these dynamics are different, the information obtained by the models is also different. In the SIR model we may be interested in the fraction of individuals (with respect to the size of susceptible people) who have caught the disease after it declines to zero, and in the case of SIS

model we might be interested in the boundary between values of the parameters in which the disease persists and those in which it does not. There is evidence that shows that the behaviour of both models occurring on complex topologies is the same as far as the existence of critical points and the nature of transition is concerned [47].

1.4.2 Synchronisation

Collective dynamics and synchronisation in large networks is a topic widely studied in different contexts, from biology to electronic circuits, and even the collective human behaviour present in some situations. In the synchronisation phenomenon many actors or components of a system adjust a given property of theirs, due to a suitable coupling configuration, or to an external force.

Synchronisation can be studied by applying the so called *master stability function* approach which was introduced for arrays of coupled oscillators and extended to the case of complex networks of dynamical systems coupled with arbitrary topologies [47, 49].

In this approach, a generic network of n coupled dynamical entities is considered. Each unit i has a m-dimensional vector field associated called x_i ruled by a local set of ordinary differential equations

$$\dot{x}_i = f_i(x_i), \tag{1.4.5}$$

which defines the equation of motion

$$\dot{x}_i = f_i(x_i) - \mathfrak{c} \sum_{j=1}^{n} \mathscr{C}_{ij} H(x_j),\ i = 1, \ldots, n \tag{1.4.6}$$

where $H : \mathbb{R}^m \to \mathbb{R}^m$ is a vectorial output function, $\mathfrak{c}$ is the coupling strength, and $\mathscr{C}_{ij} \in \mathbb{R}$ are the elements of a $n \times n$ symmetric connectivity matrix $\mathscr{C}$ with strictly positive terms in the main diagonal, this matrix specifies the strength and topology of the underlying connection linking. The coupling matrix is suitable

related with other matrices used for defining the topology of the network such as the adjacency and the Laplacian matrices [47].

The main assumption is that the network is made of identical systems, this means that the evolution function f is the same for all the nodes in the network, thus ensuring the existence of an invariant set $x_i(t) = x(t)$ for all i that represents the complete synchronisation manifold $\mathscr{S}$ [47]. For the case of $\mathscr{C}$ being symmetric, we let $\lambda_i[v_i]$ the set of real eigenvalues (of associated orthonormal eigenvectors) such that $\mathscr{C}v_i = \lambda_i v_i$. Then the zero-row condition ensures that:

- the spectrum is semi-positive: $\lambda_i \geq 0 \ \forall i$,
- $\lambda_1 = 0$ with associated eigenvector $v_1 = \pm\frac{1}{\sqrt{n}}\{1, 1, \ldots 1\}^T$ that defines the synchronisation manifold $\mathscr{S}$,
- the rest of eigenvalues λ_i, $i = 2, \ldots, n$, have associated eigenvectors v_i spanning all the other directions of the $m \times n$ dimensional phase space transverse to $\mathscr{S}$.

Master stability function arguments are used as a framework for the study of synchronised behaviours in complex networks, and mainly for gaining knowledge about the interplay between complexity in the overall topology and local dynamical properties of the coupled units. There are many situations, such as pulse-coupled networks of neurons, and the interest in these phenomena has lead to the study of non-linear coupled systems with the same aim of relating topology to the synchronisation properties of the network.

Collective behaviours in networks of ordinary differential equations have been used to model chemical reactions, ecological systems, electronic circuits, neurons and cardiac cells systems, self-organisation phenomena, travelling waves, opinion formation in social networks, consensus processes, among others [47].

1.5 Consensus

One of the fields where consensus problems in groups of experts were first formally studied is *management science and statistics*, being one of the most cited works

in this area the article "Reaching a consensus" from DeGroot, M. H. published in 1974 [24]. Another field where consensus problems have been largely studied is computer science, and from this interest, the foundations of what we now know as *distributed computing,* were established [6].

With the advent of technological means (miniaturisation of computing, communication, sensing, and actuation) that allow massive collections of data, the results from systems and control theories on consensus, have being applied to analyse, simulate and synthesise different kinds of networked systems where cooperative behaviour is an important feature for their performance [13]. These theories, together with network theory, form a powerful alternative framework from which such systems can be studied.

In networks of agents or dynamic systems, "consensus" means to reach an agreement regarding a certain amount of interest that depends on the state of all agents. A "consensus algorithm" (or protocol) is an interaction rule that specifies the information exchange between an agent and all of its neighbours on the network [7].

Agreement or consensus is one of the fundamental problems in synchronisation of complex systems, where their constituents are to agree on a joint state value. For this purpose, different models (also called algorithms or protocols) have been developed like the Axelrod [50] and the voter models [51]. In the Axelrod model it is assumed that the more similar two actors are, the more alike they are (homophily), this model shows both consensus and fragmentation into clusters as possible equilibria states [52]. The voter model is an stochastic approach that assumes that an actor has two opinions and its dynamics consists of randomly selecting one node and assigning an opinion to it, which are represented by the spins values ± 1, of one of its nearest neighbours, which in turn are selected also by random [53].

One of the works that offers a graph theoretic perspective of the problem is the one from Mesbahi and Egerstedt [3], which we mainly follow to stablish the general framework of the consensus model on simple undirected networks.

1.5.1 Consensus model and agreement state

As starting point, we consider the well know *meet-for-dinner* problem [13]. This problem formulates the situation when a group of friends decides to meet for dinner

at a particular restaurant or place, but fail to specify a precise time to meet. On the afternoon of the dinner appointment, all of the individuals realise that they are uncertain about the time the group will meet. A centralised solution for this problem would be for the group to have a conference call, to poll all individuals regarding their preferred time for dinner, and to average the times proposed to set a time when the group would meet that takes into account all the proposals.

A distributed solution to the problem would be for each individual to call, one at a time, a subset of the group. Given his/her current estimate of the time, i.e., his/her instantiation of the coordination variable, the individual may update his/her estimate of the meeting time to be a weighted average of his/her current meeting time and that of the person with whom he/she is conversing. The question is to determine under what conditions this strategy would enable the entire team to converge to a consistent meeting time. The coordination variable in the meet-for-dinner problem is the time when the group will meet. The particular time is not what is important, but rather that each individual in the group has a consistent understanding of that information.

This problem has been shown in [13] to be solved by the distributed strategy, pointing out that for such strategy to be effective, the group must act in a cooperative way so they can agree on the value of the variable of interest: they reach consensus. The information consensus guarantees that the group sharing information over a network topology have a consistent view of information critical to the coordination task. To achieve consensus, there must be a shared variable of interest, called the information state, as well as appropriate rules (algorithm or protocol) for negotiating to reach consensus on the value of that variable.

The significance of the agreement protocol is twofold. On one hand, agreement has a close relation to a host of networked problems such as flocking, rendezvous, swarming, distributed estimation, among others. On the other hand, this protocol provides a concise formalism for examining means by which the network topology dictates properties of the dynamic process evolving over it [3].

A networked system is a collection of n dynamic units that interact over an information exchange network for its operation. This network can be modelled through a graph $G = (V, E)$ where nodes represent agents and edges are inter-agent information exchange links. For the system to reach consensus, there must be an

agreement protocol for the exchange of information among the nodes trough the relative information-exchange links. The rate of change in the state of each node is lead by the sum of its relative states with respect to a subset of other nodes (neighbours).

To illustrate this situation, let the network $G = (V, E)$ be the one represented in figure 1.5.1. In this network, for example, node 1 has as neighbours nodes 2 and 3, thus it has links with both nodes meaning that there is an exchange of information between them. The relative state of this node, x_1, with respect to the state of node 2, x_2, is given by the difference $(x_2 - x_1)$. Similarly, the relative state of node 1 with respect to node 3 is $(x_3 - x_1)$. The rate of change of the states of node 1, denoted as $\dot{x_1}$, is given by the sum of the relative states of node 1 with respect to its neighbours, this is, $\dot{x}_1 = (x_2 - x_1) + (x_3 - x_1)$.

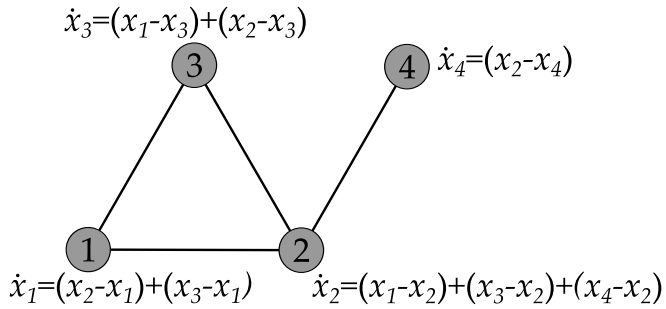


Figure 1.5.1: Illustration of the agreement protocol over a network.

Denoting the scalar state of node i as $x_i \in \mathbb{R}$, we have

$$\dot{x}_i(t) = \sum_{j \in N(i)} \left(x_j(t) - x_i(t)\right), \ i = 1, \ldots, n, \tag{1.5.1}$$

where $N(i)$ is the set of nodes neighbours of node i in the network.

If we consider a simple, undirected network, the adjacency matrix of such network is symmetric, thus we can express the rate of change of the states of node i as

$$\dot{x}_i(t) = -\sum_{j=1}^{n} a_{ij}(t) \left(x_i(t) - x_j(t)\right), \ i = 1, \ldots, n, \tag{1.5.2}$$

where $a_{ij}(t)$ is the (i, j) entry of the adjacency matrix $\boldsymbol{A}$ associated with G at time t. We can see that $a_{ij} = 0$ would denote that node i cannot exchange information with node j, otherwise $a_{ij} = 1$. Since we are considering a fixed communication topology, the values of a_{ij} are time invariant, and the overall system can be represented by

$$\dot{x}(t) = -\boldsymbol{L}(G)\, x(t), \tag{1.5.3}$$

where the positive semidefinite matrix $\boldsymbol{L}(G)$ is the Laplacian of the underlying interaction network for the nodes of the system, and $x(t) = (x_1(t), \ldots, x_n(t))^T \in \mathbb{R}^n$ is the vector of states values of the nodes at time t. We call the expression (1.5.3) the *agreement* dynamics or *consensus* model.

In this consensus model, the dynamics of each vertex in the network is pulled toward the states of the neighbouring vertices, and all the elements of the system asymptotically reach a weighted average of their initial states, which corresponds to the fixed point of their collective dynamics (see example in fig. 1.5.2). This state of agreement or consensus is formally defined as follows.

Definition 1.3. The agreement set $\mathcal{A} \subseteq \mathbb{R}^n$ is the subspace $\mathbf{span}\,\{\mathbf{1}\}$, that is,

$$\mathcal{A} = \{\, x \in \mathbb{R}^n \,|\, x_i = x_j,\ for\, all\, i,\, j \,\}\,. \tag{1.5.4}$$

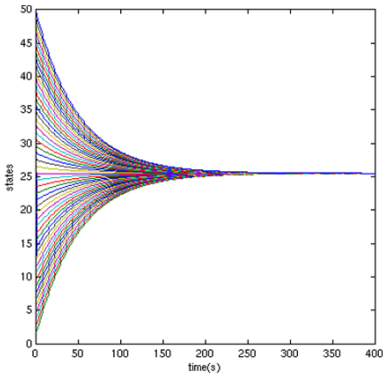


Figure 1.5.2: Example of consensus process on a star-like network with 50 nodes. Each node were assigned their number-label as initial state, i.e., node 1 has 1 as initial state value, node 2 has 2 as initial state value, and so forth. All the system reach the agreement value of 25 as consensus.

1.5.2 Convergence

We now review the mechanism by which the dynamics described on the previous page, over simple undirected networks, guides the nodes of the system to the agreement state or consensus [3].

We recall from (1.2.15) that the spectrum of the Laplacian matrix for a connected undirected network $G = (V, E)$ has the form $0 = \lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n$ with $\mathbf{1}$ as the eigenvector corresponding to the zero eigenvalue λ_1. We note that $\mathbf{L}(G)$ is symmetric and that $\mathbf{L}(G)\mathbf{1} = 0$ for an arbitrary undirected network G. Let $\mathbf{\Phi} = [\boldsymbol{\phi}_1\, \boldsymbol{\phi}_2 \cdots \boldsymbol{\phi}_n]$ be the matrix consisting of normalised and mutually orthogonal eigenvectors of $\mathbf{L}(G)$, corresponding to its ordered eigenvalues, as defined in (1.2.2.2). Furthermore, let $\Lambda = diag\,(\lambda_1, \lambda_2, \ldots, \lambda_n)$ be the diagonal matrix of eigenvalues of the Laplacian (as defined on page 32).

Using the spectral factorisation of the Laplacian, one has

$$e^{-L(G)t} = e^{-\left(\mathbf{\Phi}\Lambda\mathbf{\Phi}^T\right)t} = \mathbf{\Phi}e^{-\Lambda t}\mathbf{\Phi}^T = e^{\lambda_1 t}\boldsymbol{\phi}_1\boldsymbol{\phi}_1^T + e^{\lambda_2 t}\boldsymbol{\phi}_2\boldsymbol{\phi}_2^T + \cdots + e^{\lambda_n t}\boldsymbol{\phi}_n\boldsymbol{\phi}_n^T. \quad (1.5.5)$$

So, the solution of (1.5.3) with initial values $x(0) = x_0$ is

$$x(t) = e^{-\boldsymbol{L}(G)t}x_0 \quad (1.5.6)$$

which can be decomposed along each eigen-axis as

$$x(t) = e^{\lambda_1 t}(\boldsymbol{\phi}_1^T x_0)\boldsymbol{\phi}_1 + e^{\lambda_2 t}(\boldsymbol{\phi}_2^T x_0)\boldsymbol{\phi}_2 + \cdots + e^{\lambda_n t}(\boldsymbol{\phi}_n^T x_0)\boldsymbol{\phi}_n. \quad (1.5.7)$$

The following theorem states the convergence of the agreement protocol for static undirected networks where λ_2 plays a key role [3].

Theorem 1.4. *Let $G = (V, E)$ be a connected graph. Then the (undirected) agreement protocol* (1.5.3) *converges to the agreement set* (1.5.4) *with a rate of convergence that is dictated by $\lambda_2(\boldsymbol{L})$.*

Proof. The proof follows directly from (1.5.7) by observing that for a connected graph $\lambda_i > 0$ for $i \geq 2$; as always, $\lambda_1 = 0$. Thus

$$x(t) \to \left(\boldsymbol{\phi}_1^T x_0\right) \boldsymbol{\phi}_1 = \frac{\mathbf{1}^T x_0}{n} \mathbf{1} \tag{1.5.8}$$

as $t \to \infty$, and hence $x(t) \to \mathcal{A}$ (see fig. 1.5.3). As λ_2 is the smallest positive eigenvalue of the graph Laplacian, it dictates the slowest mode of the convergence in (1.5.8). □

As the states of the vertices evolve toward the agreement set, we have

$$\frac{d}{dt}\left(\mathbf{1}^T x(t)\right) = \mathbf{1}\left(-\boldsymbol{L}(G)\, x(t)\right) = -x(t)^T \mathbf{L}(G)\mathbf{1} = 0. \tag{1.5.9}$$

As such, the quantity $1^T x(t) = \sum_i x_i(t)$, which means, the centroid of the network states, evaluated for any $t \geq 0$, is a constant of motion for the agreement dynamics (1.5.3). Furthermore, the proof of theorem (1.4) indicates that the state trajectory generated by the agreement protocol converges to the projection of its initial state, in the Euclidean norm, onto the agreement subspace, since

$$\arg\min_{x \in \mathcal{A}} \|x - x_0\| = \frac{\mathbf{1}^T x_0}{\mathbf{1}^T \mathbf{1}} \mathbf{1} = \frac{\mathbf{1}^T x_0}{n} \mathbf{1}. \tag{1.5.10}$$

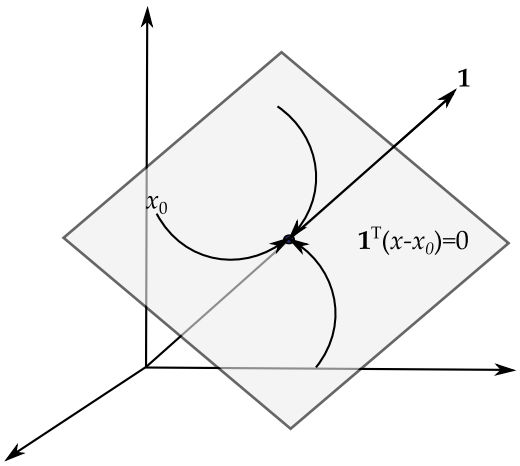


Figure 1.5.3: Illustration of the trajectory of the agreement protocol, which retains the centroid of the states of the nodes as its constant of motion.

The general form of the solution to the agreement dynamics (1.5.7), indicates that in order to have convergence to the agreement subspace from an arbitrary initial condition, it is necessary and sufficient to have $\lambda_2 > 0$. Since positivity of λ_2 corresponds to the connectivity of the network, we can conclude that the minimum order structure needed for asymptotic convergence to consensus is an interconnected network containing a spanning tree.

Proposition 1.5. *A necessary and sufficient condition for the agreement protocol* (1.5.3) *to converge to the agreement subspace* (1.5.4) *from an arbitrary initial condition is that the underlying graph contains a spanning tree.*

Considering the previous analysis, the (average) consensus value for a network under the last conditions is a convex combination of the initial states of the nodes, and we say that consensus is achieved or reached by the nodes of the network if, for all the initial states $x_i(0)$ and all $i,\ j = 1, \ldots, n$, $|x_i(t) - x_j(t)| \to 0$, as $t \to \infty$ [13].

1.5.3 Discrete time model of consensus

When the communication among the agents or actors in the network occurs at discrete instants, the information state is updated using a difference equation. The most common discrete-time consensus model is of the form [3, 6–8, 13, 15]

$$x_i(k+1) = x_i(k) + \epsilon \sum_{j \in N_i} a_{ij} \left(x_j(k) - x_i(k) \right) \tag{1.5.11}$$

where $x_i(k)$ denotes the state of node i at the communication event k, $\epsilon > 0$ is the step size and a_{ij} is defined as the $(i,\ j)$ entry of the adjacency matrix $\boldsymbol{A}$ associated with the network G.

The discrete collective dynamics of the network under this model can be written as

$$x(k+1) = \boldsymbol{P} x(k). \tag{1.5.12}$$

The matrix $\boldsymbol{P}$, know as the *Perron matrix* [7], is constructed as

$$\boldsymbol{P} = \boldsymbol{I} - \epsilon \boldsymbol{L} \tag{1.5.13}$$

where $\mathbf{I}$ is the identity matrix of order n. The following lemma from [7] states the properties of matrix $\boldsymbol{P}$:

Lemma 1.6. *Let $G = (V, E)$ be a digraph with n nodes and maximum degree $\Delta = \max_i \left(\sum_{j \neq i} a_{ij}\right)$. Then the Perron matrix $\boldsymbol{P}$ with parameter $\epsilon \in \left(0, \frac{1}{\Delta}\right]$ satisfies the following properties.*

1. $\boldsymbol{P}$ is a row stochastic non-negative matrix with a trivial eigenvalue of 1;
2. All eigenvalues of $\boldsymbol{P}$ are in a unit circle;
3. If G is a balanced graph[2], then $\boldsymbol{P}$ is a doubly stochastic matrix;
4. If G is strongly connected and $0 < \epsilon < \frac{1}{\Delta}$, then $\boldsymbol{P}$ is a primitive matrix.

Proof. Since $\boldsymbol{P} = \boldsymbol{I} - \epsilon \boldsymbol{L}$, we have $\boldsymbol{P}\mathbf{1} = \mathbf{1} - \epsilon \boldsymbol{L}\mathbf{1} = \mathbf{1}$ which means the row sums of $\boldsymbol{P}$ is 1. Moreover, 1 is a trivial eigenvalue of $\boldsymbol{P}$ for all graphs. To show that $\boldsymbol{P}$ is non-negative, notice that $\boldsymbol{P} = \boldsymbol{I} - \epsilon \boldsymbol{K} + \epsilon \boldsymbol{A}$ due to the definition of the Laplacian matrix (1.2.14), $\epsilon \boldsymbol{A}$ is a non-negative matrix. The diagonal elements of $\boldsymbol{I} - \epsilon \boldsymbol{K}$ are $1 - \epsilon k_i \geq 1 - \frac{k_i}{\Delta} \geq 0$ which implies $\boldsymbol{I} - \epsilon \boldsymbol{K}$ is non-negative. Since the sum of two non-negative matrices is a non-negative matrix, $\boldsymbol{P}$ is a non-negative row stochastic matrix.

To prove part 2, we notice that all eigenvectors of $\boldsymbol{P}$ and $\boldsymbol{L}$ are the same. Let λ_j be the jth eigenvalue of $\boldsymbol{L}$, then the jth eigenvalue of $\boldsymbol{P}$ is

$$\mu_j = 1 - \epsilon \lambda_j. \tag{1.5.14}$$

Based on Gershgorin theorem (see Appendix on page 162), all eigenvalues of $\boldsymbol{L}$ are in the disk $|s - \Delta| \leq \Delta$. Defining $z = 1 - \frac{s}{\Delta}$, we have $|z| \leq 1$ which proves part 2.

[2] A directed graph is called balanced if, for every node, its in-degree is equal to its out-degree.

If G is a balanced digraph, then $\mathbf{1}$ is the left eigenvector of $\boldsymbol{L}$, or $\mathbf{1}^T\boldsymbol{L} = 0$. This means that $\mathbf{1}^T\boldsymbol{P} = \mathbf{1}^T - \epsilon\mathbf{1}^T\boldsymbol{L} = \mathbf{1}^T$ which implies the column sums of $\boldsymbol{P}$ are 1. If we combine this result with the result in part 1, we have proved part 3.

To prove part 4, we note that if G is strongly connected, then $\boldsymbol{P}$ is an irreducible matrix [54]. To prove that $\boldsymbol{P}$ is primitive, we need to establish that it has a single eigenvalue with maximum modulus of 1. For all $0 < \epsilon < \frac{1}{\Delta}$, the transformation $\mu = 1 - \epsilon s$ maps the circle $|s - \Delta| = \Delta$ into a circle that is located strictly inside a unit disk passing trough the point $\mu = 1$. This means that only a single eigenvalue at $\mu_1 = 1$ can have modulus of 1. □

1.6 Leader-Follower Consensus

Leader-follower models were initially used for solving problems related with controlling networks of interacting agents, like mobile robots that need to perform a task in a coordinated fashion [55, 56]. The problem is approached by considering the concept of *heterogeneous networks*, which means, the set of agents in the system is partitioned into two subgroups: the group of leaders and the group of followers. The idea is that the followers execute a relatively simple and decentralised control protocol, designed with the purpose of keeping the team together, and at the same time, the leaders are assumed to have access to global information and their movements are to be defined in such a way as to take an overall performance objective into account [57].

Then, if we consider that a network has one or more nodes that drive the system to a certain state, consensus is then referred to as *leader-follower* consensus. The difference between a leaderless consensus process as stated in (1.5.3) and a consensus with one or multiple leaders is that the first converges to the average of the initial values of the states of the agents while a consensus with one leader converges to the leader state (reference state), and for a consensus with multiple leaders the system converges to the convex hull spanned by the states of the nodes acting as leaders [30], this problem is known as *controlled agreement* and is formulated in the following way [58].

If we consider a network $G = (V, E)$ with n nodes, m edges, and bidirectional

communication links having weight equal to one, the state of node i at time t is represented by $x_i(t)$ whose dynamics is described as

$$\dot{x}_i(t) = u_i(t),\ i = 1, \ldots, n, \tag{1.6.1}$$

being $u_i(t)$ the control input for node i. If we let node i to have access to the relative state of information with respect to its neighbours $N(i)$, this can be used to compute its own control input. Lets assume that the control input takes the form

$$u_i(t) = \sum_{j \in N(i)} \left(x_j(t) - x_i(t)\right),\ i = 1, \ldots,\ n, \tag{1.6.2}$$

which allows node i to access the relative state of information with respect to its neighbours and leads to the solution of the rendezvous problem [30]. This control input happens to be the expression (1.5.1) for the consensus model, so the dynamics (1.6.1) and the control input (1.6.2), can be represented together as

$$\dot{x}(t) = -\boldsymbol{L}(G)x(t), \tag{1.6.3}$$

being $x(t)$ the state vector of the system, and $\boldsymbol{L}$ the associated Laplacian $\boldsymbol{L}(G)$, which has the properties as described on page 50.

For a system with multiple stationary leaders[3], the complete set of nodes of the system is divided into two disjoint sets: the set of *leaders* $V_l \subset V$ and the set of *followers* $V_f \subset V$. Lets consider that we label all the nodes in such a way that the first $n_f < n$ represent those nodes called the followers and the rest $n_l = n - n_f$ are the leaders, being the total number of agents or actors $n = n_f + n_l$. Then we have the vector of nodes states

$$x = \begin{bmatrix} x_f \\ x_l \end{bmatrix} \tag{1.6.4}$$

[3]These nodes can also be called *anchors* or *nonconformist* agents, depending on the context the model is used.

where $x_f \in \mathbb{R}^{n_f}$ are the states of the followers and $x_l \in \mathbb{R}^{n_l}$ are the states of the leaders. Leadership designations induce a partition of the incidence matrix $\boldsymbol{B}(G)$ as

$$\boldsymbol{B}(G) = \begin{bmatrix} B_f \\ B_l \end{bmatrix}, \tag{1.6.5}$$

where $B_f \in \mathbb{R}^{n_f \times m}$, and $B_l \in \mathbb{R}^{n_l \times m}$ are the incidence matrices of the subgraphs induced by the partition of the node set V into followers and leaders. The underlying assumption of this partition, without loss of generality, is that leaders are indexed last in the original network. As a result, the correspondent graph Laplacian $\boldsymbol{L}(G)$ of the system can be partitioned as

$$\boldsymbol{L}(G) = \begin{bmatrix} L_f & l_{fl} \\ l_{fl}^T & L_l \end{bmatrix}. \tag{1.6.6}$$

where $L_f = B_f B_f^T \in \mathbb{R}^{n_f \times n_f}$, $L_l = B_l B_l^T \in \mathbb{R}^{n_l \times n_l}$ and $l_{fl} = B_f B_l^T \in \mathbb{R}^{n_f \times n_l}$.

Then, the control system is the controlled agreement dynamics, also called *leader-follower system*, where followers evolve trough the Laplacian-based dynamics

$$\dot{x}_f(t) = -L_f x_f(t) - l_{fl} u(t), \tag{1.6.7}$$

where $u(t) = x_l(t)$ is the exogenous control signal dictated by the leaders to the followers system.

Since we consider the leaders are stationary, the states of these remain unchanged and their dynamics are given by

$$\dot{u}_i(t) = 0, \ \ for\, all\, i \in V_l, \tag{1.6.8}$$

and this consideration leads us to the form of the dynamic of the controlled system expressed in matrix form as

$$\begin{bmatrix} \dot{x}_f(t) \\ \dot{u}(t) \end{bmatrix} = - \begin{bmatrix} L_f & l_{fl} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} x_f(t) \\ u(t) \end{bmatrix}. \tag{1.6.9}$$

The last expression corresponds to zeroing-out the rows of the original graph Laplacian associated with the leaders, this transformation can be accomplished via a reduced identity matrix Q_r, with zeros at the diagonal elements that correspond to the leaders, with all other diagonal elements being kept as one, this is,

$$\begin{bmatrix} L_f & l_{fl} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} = Q_r \boldsymbol{L}(G), \tag{1.6.10}$$

where

$$Q_r = \begin{bmatrix} I_{n_f} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}, \tag{1.6.11}$$

and all the zero matrices are of proper dimensions.

The following theorem states the equilibrium point for the dynamics of the leader-follower system with stationary leaders, the proof of this theorem is not included here but can be consulted in [59].

Theorem 1.7. *Given fixed leader positions x_l, the equilibrium point under the follower dynamics in* (1.6.7) *is*

$$x_f = -L_f^{-1} l_{fl} x_l, \tag{1.6.12}$$

which is globally asymptotically stable.

As a result of the previous theorem, and considering that all leaders are stationary, the followers would asymptotically approach the equilibrium point

$$x_f^e = -L_f^{-1} l_{fl} x_l. \tag{1.6.13}$$

Since x_f^e is an equilibrium, we have that

$$\dot{x}_i^e = 0 = -\sum_{j \in N(i)} \left(x_i^e - x_j^e\right) \tag{1.6.14}$$

for all nodes followers[4]. This means that

$$x_i^e = \frac{1}{|N(i)|} \sum_{j \in N(i)} x_j^e, \tag{1.6.15}$$

and this is, the equilibrium point x_i^e for a node follower i lies in the convex hull spanned by its neighbours (may the be leaders or followers).

Now, considering the complete system, if every follower ends up in the convex hull spanned by its neighbours, and the only nodes who do not need to satisfy this are the leaders, every follower will end up in the convex hull, denoted by Ω_l, spanned by the leaders (see fig. 1.6.1). This result was established and proved in [30, 60], and formalised in the following lemma:

Lemma 1.8. *Given a connected, static network topology with multiple static leaders, the followers will asymptotically end up in the convex hull spanned by the leaders, i.e.*

$$x_i^e \in \Omega_l, \;\; i = 1, \ldots, n_f. \tag{1.6.16}$$

[4]The notation implies that if node j is a leader, then x_j^e represents its static position, even though, strictly speaking, is not an equilibrium point, but a static input.

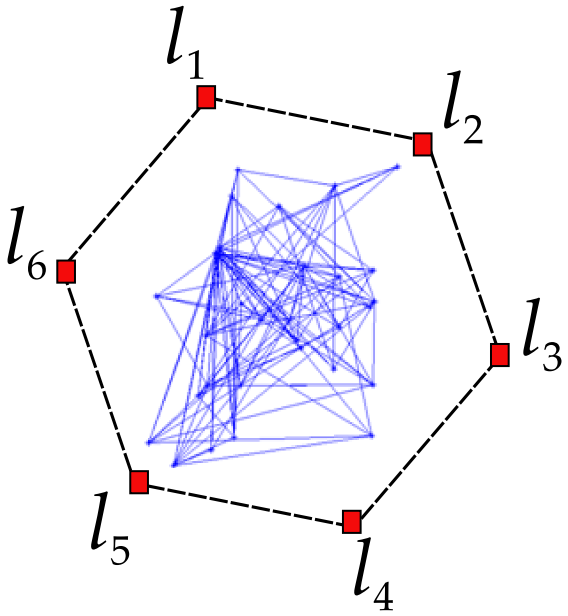


Figure 1.6.1: Convergence of followers (small blue dots) to the convex hull spanned by the leaders' states (red squares) in a network with six leaders. The convex hull is shown by the discontinuous line segments connecting the leaders. The lines between followers denotes edges.

Summary

The aim of this chapter was to briefly cover two main aspects for the study of networks: their structure, and their dynamics. We started by defining some basic concepts from graph theory which will be used through this work. Regarding the dynamics on networks, we have briefly defined what a dynamical system is, and we have showed some examples of the possible dynamics that can modelled on a network, such as an epidemic spreading process. Finally we have dedicated more extensive part to define the basis of the consensus model, its convergence conditions and its variant to take into account the case when some actors take a leadership role. These concepts will serve as basis for our generalised leader-follower consensus model and the analysis of the impact of peer-pressure in the system. All the theory presented can be studied more deeply by going into the references provided.

In the following chapter we will present a review of some of the most known works on distributed algorithms, the agreement problem, multiagent systems, and cooperative control that paved the way to what is currently known as distributed control for multi-agent systems, and from which, the consensus model used in this thesis, comes from.

Chapter 2

CONSENSUS AND COOPERATIVE CONTROL IN NETWORKED SYSTEMS - A REVIEW

Consensus models have been studied by different areas, but perhaps distributed computation was one of the first ones where consensus-like models were used for tackling problems where cooperative behaviour among different parts of a system is needed. The recent availability of big amounts of data, an increasing interest in controlling unmanned vehicles, and the development of network science framework have contributed to a new wave of research dealing with cooperative behaviour in complex systems, where consensus has attracted the attention of many researches. We review some of the most important works that provided the grounds for what we now call cooperative control of networked systems (see figure 2.0.1). We start by touching the roots with the definition of distributed algorithms, following by some works on the agreement problem in multiagent systems, and finally the main theme of the thesis: the consensus protocol in networks, which is the result of applying the theory of distributed algorithms and agreement problems, to multiagent networked systems. The knowledge coming from these contributions is being used for studying different kinds of complex systems, such as physiological systems, large-scale energy systems, fleets of vehicles, diffusion of information in

online social networks, to mention a few. Here network science offers a very handy set of tools that has been playing an important role in modelling, simulating and analysing such kind of systems.

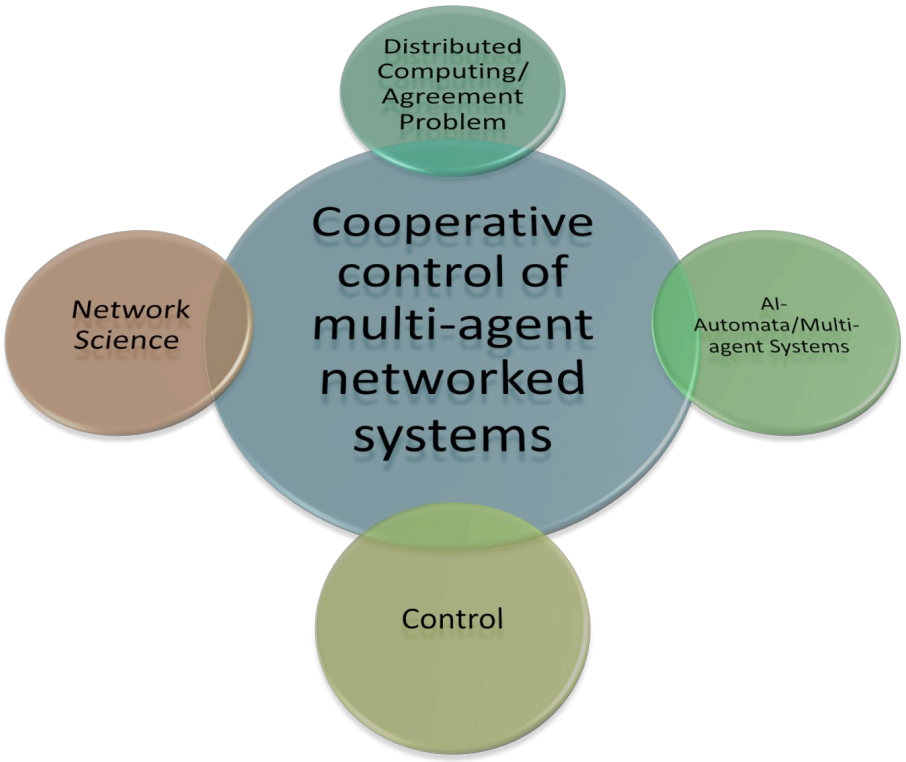


Figure 2.0.1: Main areas of knowledge contributing to the cooperative control of multiagent networked systems field.

2.1 Distributed algorithms

Problems of reaching an agreement between different parts of a system have been discussed among different areas, but one of the first disciplines where these problems arouse was related with distributed algorithms, a research area that has been called *Distributed Computing,* which grounded the framework for studying distributed systems [61].

The fundamental problem here is to develop algorithms that let a set of processes to seek cooperative behaviour on some common task, this is due, in modern computing, a program usually is executed on several processes [62]. To cooperate on

some common task, the processes need to exchange messages or information using some communication network. From this perspective, processes are considered as abstractions that may represent different elements like a computer, a processor within a computer, or a thread of execution (a sequence of instructions that may execute in parallel with each other) within a processor. The major difficulty in the area is to design an algorithm that allows a system to reach a robust kind of cooperation, despite failures or disconnections of some processes that are inherent to most distributed environments. When a subset of processes fails or got disconnected, all those processes that are still operating have the challenge to synchronize their activities in such a way that remain consistent, or in other words, the cooperation process must allow to tolerate partial failures.

As a mean for designing distributed algorithms, the underlying physical system is abstracted as a relation of two sets: processes which are the active entities that perform computations, and links, being the exchanging of messages trough some physical or logical communication network. Describing the relevant components on an abstract way, identifying their intrinsic properties, and characterising their interactions leads to what is call a *system model* [62]. Thus, given a system model, the aim is to build abstractions that capture recurring interaction patterns in distributed applications. This kind of cooperation among processes can often be modelled as a distributed agreement problem and its corresponding proposes of solution are called distributed algorithms. A distributed algorithm is conceived as a collection of distributed automata (one automaton per process). A definition of what an automaton is would be: [63]

Let $\mathfrak{Q}$ and $\mathfrak{A}$ be non-empty finite sets. A finite automaton is a 5-tuple $\mathbb{A} = (\mathfrak{Q}, \mathfrak{A}, \mathfrak{f}, \mathfrak{q}_0, \mathfrak{F})$ where:

1. $\mathfrak{Q}$ is a finite set of states.
2. $\mathfrak{A}$ is a finite set of input alphabet.
3. $\mathfrak{f} : \mathfrak{Q} \times \mathfrak{A} \to \mathfrak{Q}$ is the transition function.
4. $\mathfrak{q}_0 \in \mathfrak{Q}$ is the start or initial state.
5. $\mathfrak{F} \subseteq \mathfrak{Q}$ is the set of accepting or final states.

Where, for each possible combination of state and input symbol, the transition function $\mathfrak{f}$ specifies exactly one next state. It is important to note that the automaton $\mathfrak{A}$ must have at least one initial state, but for its set of final states can be empty, in which case it is said that the automaton has no accepting states. A way of representing an automaton is by a graph called state diagram, as can be seen in figure 2.1.1.

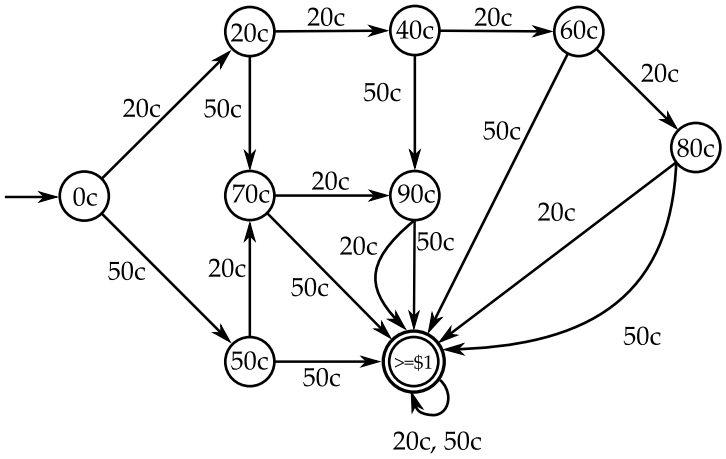


Figure 2.1.1: State diagram of a simple vending machine to represent the process of receiving coins. All possible states are represented by nodes and the possible ways for reaching them are represented by directed links. The initial state here is represented by node 0c with an arrow to its left side, while the final state is represented by a double circled node.

An automaton regulates the way that a process executes its computational steps, i.e. how it reacts to a message. The behaviour of each peer, characterized by the set of messages that is capable of producing and accepting, the format of each message, and the legal or accepted sequences of messages, is called protocol. The purpose of a protocol is to guarantee the execution of a distributed algorithm. It is not assumed that processes have access to any sort of physical clock but, however, it is still possible to measure the passage of time on the transmission and delivering of messages: time is defined with respect to communication (the moment when the message is produced and received) and is called logical time.

Then, as the aim is to build appropriate abstractions, the protocol must succeed in capturing the recurring interaction patterns in distributed applications, this task has been reached by the protocol for distributed agreement or consensus, which has been used to model situations where the parts of a system need to agree on a value related to a variable of interest.

2.2 Agreement problem and multi-agent systems

This problem was first defined in 1982 as "The Byzantine Generals Problem" [64] where the problem of handling malfunctioning components in computer systems were modelled in terms of a group of generals of the Byzantine army camped with their troops around an enemy city. Communication was only trough a messenger, and the generals were required to agree upon a common battle plan. However, it was considered that one or more of these generals could be traitors and would look to confuse the others, then the problem was to find an algorithm to ensure that the loyal generals would reach agreement. The use of graphs in analysing different solutions for this problem was a constant.

Later Lynch in 1996 [61] published a work on distributed algorithms where merged different concepts in a general framework and used the concept of network to denote the abstraction of the systems studied, applying graph theory tools for analysing distributed algorithms. In this work, the consensus problem is also indistinctly refereed to as the agreement problem, and is formulated as follows:

Assume that the system or network is an n-node connected undirected graph with processes $1, \ldots, n$, where each process knows the entire graph. Each process starts with an input from a fixed value set $\mathbb{V}$ in a designated state component; also assume that, for each process, there is exactly one start state containing each input value. The goal for the processes is to eventually output decisions from the set $\mathbb{V}$, by setting special decision state components to values in $\mathbb{V}$. Links were assumed to be perfectly reliable, i.e. all the messages that were sent were also delivered.

Agreement problems must satisfy some properties: [62]

- Termination: every correct process eventually decides some value.

- Validity: If a process decides x, then x was proposed by some process.
- Integrity: No process decides twice.
- Agreement: no two correct processes decide differently.

So, each process has an initial value that proposes for the agreement, trough the first, or primitive, proposition. The primitive values are provided to the processes and the act of proposing values is local. This act activates a transmission of events through which the processes exchange their proposed values in order to eventually reach an agreement or consensus.

Distributed algorithms, and mainly agreement processes models were later used in artificial intelligence for analysing the dynamics of autonomous agents. One definition of what an autonomous agent represents was given by Franklin and Gasser in 1997 [65]: "An autonomous agent is a system situated within and as part of an environment that senses that environment and acts on it, over time, in pursuit of its own agenda and so as to effect what it senses in the future."

The definition of agent points out an intrinsic characteristic of this object: its ability to act on its own environment, which leads to the property of being able to interact with other agents within the same system. Here, the analysis of dynamic behaviour of different complex systems conceptualised as multi-agent systems is achieved by applying the network systems framework to model the underlying interactions among agents. Multi-agent systems are characterized by a set of autonomous agents, each with local information, and the ability to perform an action looking to coordinate their own decisions with the decisions taken by the rest of the agents, in order to achieve a desired global behaviour. Then, consensus is a suitable tool to model and simulate the collective behaviour of this particular kind of systems where reaching an agreement is important for their operation.

Automatic negotiation is useful in multi-agent systems as it provides a method of aggregating distributed knowledge. That is, in a problem where each agent has different local knowledge, negotiation can be an effective method for finding the desired global course of action which maximizes the utility without having to aggregate all local knowledge in a central location. This metaphor of autonomous agents cooperating in such a way, looking to solve a problem that cannot be solved

by any one agent by itself (due to limited abilities or knowledge) was the central metaphor from which the field of distributed artificial intelligence, later known as multi-agent systems, appeared [66].

Another work on agents states that the aim of multi-agent systems is to find methods that make possible to build complex systems composed by autonomous agents who, despite of operating on local knowledge and having limited abilities, are capable of reaching the desired global behaviours [67].

A common assumption about preferences of agents is that they are captured by an utility function that provides a map from the states of the world (all possible states) to a real number. Thus, if we let $\mathbb{V}$ be the set of states in the world the agent can perceive, then the utility function of agent i can be stated as

$$u_i : \mathbb{V} \to \mathbb{R} \tag{2.2.1}$$

The states are defined as those pertaining to the world and that the agent can perceive. Given an utility function for an agent, an order of preference can be defined for the choices of the agent over the states of the world, by comparing the utility values of two states so it can be determined which one is preferred by the agent. This preference order has the following properties:

- Reflexive: $u_i(s) \geq u_i(s)$.
- Transitive: If $ui(a) \geq ui(b) \text{and} ui(b) \geq ui(c)$ then $ui(a) \geq ui(c)$.
- Comparable: $\forall a, b \text{either } u_i(a) \geq u_i(b)$ or $u_i(b) \geq u_i(a)$.

The utility functions can be used to describe the behaviour of agents and also is useful for capturing the various trade-offs that an agent must make, along with the value or expected value of its actions.

2.3 Consensus in networked systems

Convergence to a common value regarding a certain variable of interest among the components of a system is called consensus or agreement. This model has

been proposed as a solution to coordination and distributed control of complex systems where cooperative behaviour is the central paradigm, thus the concept of cooperative control has been coined among the scientific community to referrer to this kind of problems. Cooperative control is concerned with systems that can be characterised as a collection of decision-making components with limited processing capabilities, locally sensed information, and limited inter-component communications, all seeking to achieve a collective objective [68]. The primary distinguishing feature of a cooperative control system is distribution of information, and a secondary feature is complexity. The recent boom on research motivated by the application of mobile unmanned systems operating with either complete autonomy or semi-autonomy has been reflected in the number of works on the area.

We can trace the origins of the recent interest on modelling distributed cooperative behaviour back to Reynolds' work [69]. He proposed a model for simulating the aggregate motion of a flock of birds, a herd of land animals or a school of fishes. The way he approached the problem was based on considering every animal being a particle (called *boid*) and all the group's motion being a distributed behavioural model where each particle would be an independent actor navigating according to its local perception of the dynamic environment, thus the aggregate motion of the entire flock, herd or school was the result of the interaction of the relatively simple behaviours of the individual simulated animals or boids ruled by three principles (see fig. 2.3.1):

1. Obstacle avoidance: avoid collisions with nearby flockmates.
2. Alignment/velocity matching: attempt to match alignment/velocity with nearby flockmates.
3. Flock cohesion: attempt to stay close to nearby flockmates.

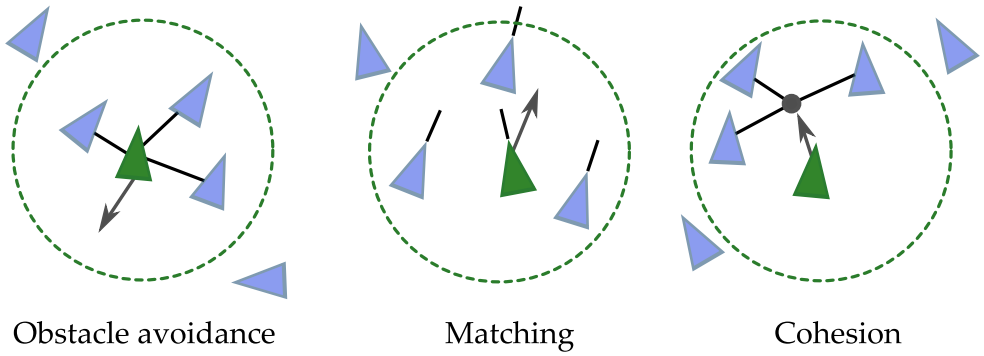


Figure 2.3.1: Illustration of the rules for Reynolds model. The boids try to: (left) avoid obstacles within certain radius (centre) match their alignment/velocity to other boids within certain radius (right) stay close to other boids within certain radius

Later, Vicsek et al. proposed a discrete mathematical model of n autonomous agents moving in the plane with the same time speed but different headings (directions) following the work of Reynolds. The only rule of the model was "at each time step a given particle driven with a constant absolute velocity assumes the average direction of motion of the particles in its neighbourhood of radius r with some random perturbation added" (see fig. 2.3.2) [70]. They provided results from computational simulations which demonstrated that the proposed nearest neighbour rule lead all agents to eventually move in the same direction despite the absence of centralised coordination and despite the fact that each agent's set of nearest neighbours changed with time as the system evolved.

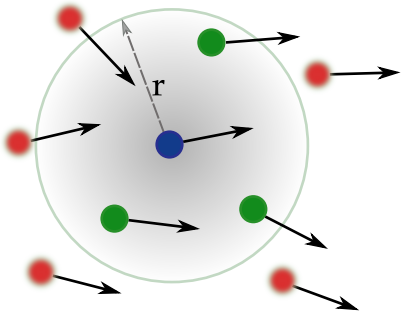


Figure 2.3.2: Illustration of Vicsek model. Any given particle (blue dot) assumes the average direction of motion of the particles in its neighbourhood of radius r (green dots). Those particles out of the neighbourhood (red dots) do not make any influence.

Although the concept of consensus was previously studied by DeGroot [24], the modern theoretical framework, including graph theoretic methods proposed for solving consensus problems for networked dynamic systems, was introduced by Olfati-Saber and Murray [15], following the PhD dissertation of Fax [71], and other previous works of Fax and Murray [12], Jadbabaie et al. [72], Mesbahi. [73], and Olfati-Saber [74]. These works, containing different concepts from distributed algorithms, consensus protocols, graph theory, multi-agent systems and control theory, were among the first modern attempts to show a multidisciplinary ordered set of tools for analysing the distributed control of complex systems. From these publications, one of the first general definitions of consensus (and its protocol) in networked systems was given:

"In networks of agents or dynamic systems, *consensus* means to reach an agreement regarding a certain amount of interest that depends on the state of all agents. A *consensus algorithm* (or protocol) is an interaction rule that specifies the information exchange between an agent and all of its neighbours on the network" [7].

Shamma compiled different works on cooperative control of networked systems including the formal definition of consensus previously given by Olfati-Saber and Murray [68]. In this work it is stated that one of the main motivations for the growing number of research groups studying cooperative control of systems was a (U.S.) National Research Council committee's study on Network Science for

Future Army Applications in 2005, which stated that the Army asked the NRC to conduct a study to define advanced operating models and architectures for future Army laboratories and centres focused on network science, technologies, and experimentation (NSTE). Some of the members of the committee were: Albert-Laszlo Barabasi, Richard M. Murray and Duncan J. Watts, and from this, the close link between cooperative control and network science on recent years can be explained.

Mesbahi in previous works pointed out that the graph Laplacian and its spectral properties were important concepts that played a central role in convergence analysis of consensus and alignment algorithms, reinforcing the idea that the interaction geometry or structure of the systems has an important position in the analysis and synthesis of networked multi-agent systems. He also had proposed the dynamic extension of the theory of graphs as a framework to model distributed dynamic systems where the topology of the interaction among its elements evolves in time [73].

Other works applying graph theoretic techniques for mathematical analysis of emergent behaviours were those from Fax and Murrray [75] and Desai et al. [76]. They defined the notion of an average heading vector in terms of graph Laplacians, and showed that this idea lead to the Vicsek et al. model as well as to other decentralised control models. In their approach, they modified Vicsek et al. model by adding one agent labelled as 0, which acted as the group's leader, moving at the same constant speed but with fixed heading θ_0.

There was an explosion of publications related with distributed cooperative control of multiagent systems from the early 2000's, a vast review of such works that provide a broad picture about consensus problems in networks is given by Mesbahi and Egerstedt [3], Olfati-Saber, Fax and Murray [7], Ren, Beard and Akins [8], Ren and Beard [13], and Kocarev [6] which summarise the results from many previous works and give a compact set of theoretical tools for boarding this research question.

In order to understand the role of cooperation in performing coordinated tasks, it has been pointed out that we need to distinguish between unconstrained and constrained consensus problems [7]. An unconstrained consensus problem is simply the alignment problem in which it suffices that the state of all agents asymptotically

be the same. In contrast, in distributed computation of a function $f(z)$, the state of all agents has to asymptotically become equal to $f(z)$, meaning that the consensus problem is constrained. Olfati-Saber, Fax and Murray refer to this problem as the f-consensus problem. The authors state that solving the f-consensus problem is a cooperative task and it requires the willing participation of all the agents. So, cooperation can be interpreted as "giving consent to providing one's state and following a common protocol that serves the group objective".

The general consensus model on networks was proposed on the works of Olfati-Saber and Murray [7]: the interaction topology of agents is represented using a directed graph $G = (V, E)$, with the set of nodes $V = \{1, 2, \ldots, n\}$ and edges $E \subseteq V \times V$. The neighbours of agent i are denoted by $N_i = \{j \in V : (i, j) \in E\}$. According to the authors, a simple consensus algorithm that looks to reach an agreement regarding the state of n integrator agents with dynamics $\dot{x_i} = u_i$ can be expressed as a nth-order linear system on a graph

$$\dot{x}_i(t) = \sum_{j \in N_i} (x(t)_j - x_i(t)) + b_i(t),\ x_i(0) = z_i \in \mathbb{R},\ b_i(t) = 0, \tag{2.3.1}$$

and the collective dynamics of the group following the last protocol is written as

$$\dot{x} = -Lx, \tag{2.3.2}$$

where $L = [l_{ij}]$ is the graph Laplacian of the network and its elements are defined as

$$l_{ij} = \begin{cases} -1 & j \in N_i \\ |N_i| & j = i \end{cases}, \tag{2.3.3}$$

being $|N_i|$ the number of neighbours of node i (or the out-degree of node i).

They stated that an equilibrium of the system (2.3.2) is a state in the form $x^* = (\alpha, \ldots, \alpha)^T = \alpha\mathbf{1}$ where all nodes agree, and they showed that x^* is a unique equilibrium of (2.3.2) for connected graphs. Moreover, the consensus value

is $\alpha = \frac{1}{n}\sum_i z_i$, that is, equal to the average of the initial states. This implies that, irrespective of the initial states of each agent, all agents reach an asymptotic consensus regarding the value of the function $f(z) = \frac{1}{n}\sum_i z_i$. The role of protocol (2.3.1) is to provide a systematic consensus mechanism in large networks to compute the average $f(z)$.

The iterative version of a consensus model corresponding to the discrete case of system (2.3.1) was formulated as

$$\pi(k+1) = \pi(k)P, \tag{2.3.4}$$

being $P = I - \epsilon L$, I is the identity matrix of corresponding dimension, and $\epsilon > 0$. In this model, the ith element of the row vector $\pi(k)$ denotes the probability of being in state i at iteration k. For any arbitrary graph G with Laplacian matrix L and sufficiently small ϵ, the matrix P satisfies the property $\sum_j p_{ij} = 1$, with $p_{ij} \geq 0, \forall i, j$. Hence P is a transition probability matrix for (2.3.4), which is a Markov chain.

Matrix P has been referred to as Perron matrix of graph G with parameter ϵ [7], and its properties were stated in the following lemma:

Lemma 2.1. *Let G be a directed graph with n nodes and maximum degree $\Delta = max_i\left(\sum_{j\neq i} a_{ij}\right)$. Then, the Perron matrix P with parameter $\epsilon \in \left(0, \frac{1}{\Delta}\right]$ satisfies the following properties:*

i) P is row stochastic nonnegative matrix with a trivial eigenvalue of 1.

ii) All eigen values of P are in a unit circle.

iii) If G is a balanced graph, then P is a doubly stochastic matrix.

iv) If G is strongly connected and $0 < \epsilon < \frac{1}{\Delta}$, then P is a primitive matrix.

Olfati-Saber and Murray pointed out that the condition $\epsilon < \frac{1}{\Delta}$ in iv) is necessary, and that if an incorrect step-size is used, then P would not longer be a primitive matrix because it could have multiple eigenvalues of modulus 1.

Another important result was the convergence analysis of the discrete-time consensus model, which relies on the lemma:

Lemma 2.2. *(Perron-Frobenius) Let P be a primitive nonnegative matrix with left and right eigenvectors w and v, respectively, satisfying $Pv = v$, $w^T P = w^T$, and $v^T w = 1$. Then $lim_{k\to\infty} P^k = vw^T$.*

The convergence and group decision properties of the iterative consensus model with stochastic Perron matrix was stated in the following result: [7]

Theorem 2.3. *Consider a network of agents $x_i(k+1) = x_i(k)+u_i(k)$ with topology G applying the distributed consensus algorithm*

$$x_i(k+1) = x_i(k) + \epsilon \sum_{j\in N_i} a_{ij} \left(x_j(k) - x_i(k)\right),$$

where $0 < \epsilon < \frac{1}{\Delta}$, and Δ is the maximum degree of the network. Let G be a strongly connected digraph. Then

i) A consensus is asymptotically reached for all initial states.

ii) The group decision value is $\alpha = \sum_i w_i x_i(0)$ with $\sum_i w_i = 1$.

iii) If the digraph is balanced (or P is doubly-stochastic), an average-consensus is asymptotically reached and $\alpha = \frac{\left(\sum_i x_i(0)\right)}{n}$.

Mesbahi and Egerstedt [3] showed that when the notion of adjacency in a network is symmetric, the overall system can be represented as in equation (2.3.2), but when the adjacency is asymmetric, the correspondent expression for the agreement dynamics becomes

$$\dot{x}(t) = -L(D)x(t), \tag{2.3.5}$$

here D is the underlying directed interconnection, that is, the weighted digraph of the network. In both cases, symmetric and asymmetric, the dynamics of each vertex is "pulled" toward the states of neighbouring vertices, so all vertices would asymptotically reach some weighted average of their initial states, which corresponds to the fixed point of their collective dynamics. Such state of agreement,

called the agreement set, is defined as the set $A \subseteq \mathbb{R}^n$ which is the subspace $\mathbf{span}\{\mathbf{1}\}$, that is, $A = \{x \in \mathbb{R}^n | x_i = x_j, \forall i, j\}$.

They proved that for a connected undirected graph, the convergence rate of the consensus model was related with the smallest positive eigenvalue of the graph Laplacian and so this value would dictate the slowest mode of convergence. They also stated as necessary and sufficient condition that this eigenvalue would be positive in order to have a convergence to the agreement subspace from an arbitrary initial condition. As positivity of the second eigenvalue of the Laplacian corresponds to the connectivity of a graph G, then they concluded that the minimum order structure needed for asymptotic convergence to agreement is an interconnected network containing a spanning tree.

These authors generalised the convergence analysis for the agreement protocol done for undirected and unweighted graphs, to those weighted and directed ones, and provided the following theorem for the convergence of the agreement protocol for a digraph:

Theorem 2.4. *For a digraph D containing a rooted out-branching, the state trajectory generated by* (2.3.2) *initialised from x_0, satifies*

$$lim_{t\to\infty} x(t) = \left(p_1 q_1^T\right) x_0,$$

where p_1 and q_1 are, respectively, the right and left eigenvectors associated with the zero eigenvalue of $L(D)$, normalised such that $p_1 q_1^T = 1$. As a result, one has $x(t) \to A$ for all initial conditions if and only if D contains a rooted out-branching.

They, observed that if a digraph contains a rooted out-branching and is balanced, then the common value reached by the agreement protocol is the average value of the initial nodes, that is the average consensus. This remark lead to the following theorem:

Theorem 2.5. *The agreement protocol over a digraph reaches the average consensus for every initial condition if and only if its weakly connected and balanced.*

There are many problems involving interconnection of dynamic systems in various areas of science and engineering and despite their differences, inherent to the proper

areas of application, they are closely related to consensus problems for multi-agent systems. Such kind of problems were divided as: [7]

1. Synchronization of Coupled Oscillators: The problem of synchronization of coupled oscillators has been studied by many scientists from different fields such as physics, biology, neuroscience, and mathematics. The most known model for this phenomenon was proposed by Kuramoto in 1975 [77], this model is now known as the Kuramoto model. The authors considered the generalized Kuramoto model of coupled oscillators on a graph with dynamics

$$\theta_i = k \sum_{j \in N_i} sin\left(\theta_j - \theta_i\right) + \omega_i,$$

where θ_i and ω_i are the phase and frequency of the ith oscillator. This model is the nonlinear extension of the consensus model in (2.3.1) and its linearisation around the aligned state $\theta_1 = \cdots = \theta_n$ is identical to system (2.3.2) plus a nonzero input bias $b_i = \frac{(\omega_i - \bar{\omega})}{k}$ with $\bar{\omega} = \frac{1}{n}\sum_i \omega_i$ after a change of variables $x_i = \frac{(\theta_i - \bar{\omega}t)}{k}$.

2. Flocking Theory: Different works on flocking theory have been developed, these behaviour has been studied on animals, people and mobile agents [69, 78–80]. Flocks of mobile agents equipped with sensing and communication devices can serve as mobile sensor networks for the task of distributed sensing in an environment. A theoretical framework for design and analysis of flocking algorithms for mobile agents with obstacle-avoidance capabilities was developed by Olfati-Saber [81]. The role of consensus algorithms in particle-based flocking is related with the situation where an agent is looking to match its velocity with respect to its neighbours. In this work, the author demonstrated that flocks are networks of dynamic systems with a dynamic topology. This topology is a proximity graph that depends on the state of all agents and is determined locally for each agent, i.e., the topology of flocks is a state-dependent graph.

3. Fast Consensus in Small-Worlds: The design of networks for achieving faster consensus algorithms has attracted important attention from researchers that try to to accomplish this characteristic in a network. This goal leads to look for ways of increasing the connectivity of a network which is a measure of

speed of convergence of consensus algorithms. [82–85]. One approach is to keep the weights of the links fixed and design the topology of the network in such a way that allows achieving high algebraic connectivity. In [86] a randomized algorithm was proposed for designing this kind of networks, based on the random rewiring idea of Watts and Strogatz and their small-world model [34]. The random rewiring of existing links of a network would give rise to considerably faster consensus algorithms. This is due to multiple orders of magnitude increase in algebraic connectivity of the network in comparison to a lattice type nearest-neighbour graph.

4. Rendezvous in Space: Another form of consensus problems is rendezvous in space [30, 87–91]. This is equivalent to reaching a consensus in position by a number of agents with an interaction topology that is position induced (i.e., a proximity graph) [92]. This type of rendezvous is an unconstrained consensus problem that becomes challenging under variations in the network topology [93]. Flocking is somewhat more challenging than rendezvous in space because it requires both inter-agent and agent-to-obstacle collision avoidance.

5. Distributed Sensor Fusion in Sensor Networks: Another application of consensus problems is the distributed sensor fusion in sensor networks [8, 94, 95], as these systems have become cheaper, the problem of allocating resources within a network, in a distributed fashion, is matter of design and implementation [96]. This process of allocating sensors in this way it has been done by multi-agent systems techniques by posing various distributed averaging problems which requires implementing a Kalman filter, approximate Kalman filter or linear least-squares estimator as average-consensus problems [97]. Low-pass and high-pass consensus filters are also developed to dynamically calculate the average of their inputs in sensor networks.

6. Distributed Formation Control: This problem is one of the best studied by the Engineering field [12, 76] as multivehicle systems are considered a special category of networked systems that is important due to their commercial and military applications. There are two main approaches to distributed formation control:

   (a) Representation of formations as rigid structures and the use of gradient-

based controls obtained from their structural potentials [71, 98].

(b) Representation of formations using the vectors of relative positions of neighbouring vehicles and the use of consensus-based controllers with input bias [99–102].

The second approach is related with consensus problems as moving in formation is a cooperative task and requires consent and collaboration of every agent in the formation.

2.4 Controllability and networks

With the advent of network science, there has been an increasing interest in the study of complex dynamical systems and their control as a means for guiding or forcing the network to achieve a desired behaviour, i.e. reaching a desired state. In practice, controlling every single node might be impossible for a network with huge number of nodes, and also this task might be unnecessary [103]. Then, diverse control strategies have been proposed to achieve the goal of controlling a network. Among these strategies, the so called pinning strategy offers a way to control an entire systems by adding control inputs to a fraction of selected nodes.

The topic of controllability is not new, and comes from the initial work of Lin in 1974 [104] where the concept of structural controllability was introduced for a linear time-invariant system of the form

$$\dot{x} = Ax + bu$$

with $x \in \mathbb{R}^n$, $u \in \mathbb{R}$. The matrices $A \in \mathbb{R}^{n\times n}$ and $b \in \mathbb{R}^n$ are assumed to have compatible dimensions and time invariant. Then the author defined the graph of the systems:

Given a pair (A, b), the graph G of this system is that one which contains exactly $n + 1$ nodes, v_1, $v_2, \ldots,$ v_{n+1}, and all of whose edges are obtained as follows: For every non-zero entrey c_{ij} of the $n \times (n + 1)$ matrix (Ab), the graph contains the

oriented edge $(v_{j,}v_i)$ (an arrow going from v_j to v_i). The node v_{n+1}, which corresponds to the $n+1$th column of (Ab), will be called the origin of G. The set which contain all the nodes of G was called the vertex set of G. For every oriented edge (v_j, v_i) in G, the node v_j was called the origin of this edge; a node v in the vertex set of G was called the final node if v was not the origin of any oriented edge in G.

The property of controllability of the pair (A, b) was reduced to a property of the graph representing the system: "a system is structurally controllable if and only if the graph of the system is spanned by a *cactus*" [104]. The structure called cactus was the key structure for his definition of controllability (see fig. 2.4.1).

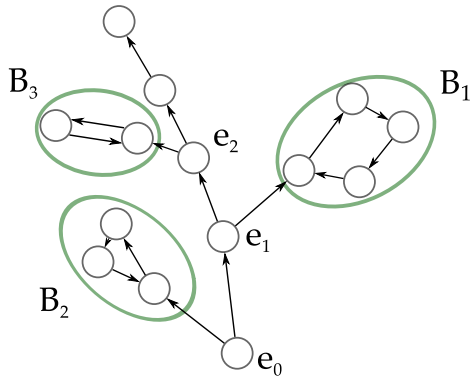


Figure 2.4.1: Illustration of Lin's proposed *cactus* structure. A cactus is conformed of *stems* and *buds*. Every bud (marked with a letter B) has an associated initial node-stem (marked with a letter e).

Later, Tanner [56] derived necessary and sufficient conditions for a group of systems interconnected via nearest neighbour rules, to be controllable by one of them acting as a leader. He stated that connectivity seemed to have an adverse effect on controllability.

Different works studying the controllability problem for multi-agent systems have appeared in recent years [16, 17, 30, 59, 60, 71, 72, 105–113], the resurgence of this topic was boosted by a work of Liu et al. [14] on controllability of complex networks, where the starting of a framework on controlling complex self-organised

systems was proposed. They stated in their work that the number of driver nodes is determined mainly by the network's degree distribution, and showed that sparse inhomogeneous networks are the more difficult to control according to their approach. They also pointed out that the driver nodes tend to avoid the high-degree nodes. These authors used the concept of structural controllability from Lin to build their analysis.

The work of Lui et al. has stimulated the development of different studies of this topic [20,103,114–126], leading to many discussions on the possibility of controlling complex systems, but the debate is still open by the time we write these lines.

Summary

We have presented some of the works that have contributed to the current field of cooperative control of multiagent systems. We started by the roots, which were established by the area of distributed computing, which studied the problem of designing algorithms that allow computational processes to work in a cooperative way. This problem was latter formulated as an agreement problem among automata, and due to its interacting nature, the concept of a network of automata was introduced. These works from distributed algorithms were later used by the field of artificial intelligence to study the dynamics of multiagent systems trying to reach an agreement during the execution of common tasks. Finally, the studies on the agreement problem were used as a solution to the more recent problem of coordinating and distributively controlling systems where cooperative behaviour is the central paradigm, thus the concept of cooperative control was coined.

Part II

Results and Discussion

Chapter 3

CONSENSUS, LEADERSHIP AND DIFFUSION OF INNOVATIONS UNDER PEER PRESSURE IN SOCIAL NETWORKS

The social group's pressure on an individual - peer pressure (PP) - has attracted the attention of scholars in a variety of disciplines, spanning sociology, economics, finance, psychology, and management sciences [127–130]. In analysing PP we should consider not only those individuals directly linked to a particular person, but also those who exert indirect social influence over other persons as well [131–134]. Although PP is an elusive concept, it can be considered a decreasing function of a given individual's socio-cultural distance from the group. Thus, an individual's opinion may be influenced more strongly by the pressure exerted by those socio-culturally closer to her.

The main question we deal with in our work is: What is the effect of the combined direct and indirect social influences - peer pressure - on a social group's collective decisions? One of the main contributions of our work is the development of a model that captures PP as a function of the socio-cultural distance between individuals

in a social group. The contents of this chapter come from our published work [135].

Our model builds upon the concept of consensus on social networks which is well documented across the social sciences, with examples ranging from behavioural flocking in popular cultural styles, emotional contagion, collective decision making, pedestrians' walking behaviour, and others [7, 9–11].

3.1 Consensus under peer pressure

3.1.1 Consensus with leaders in social networks

For modelling consensus on social networks, we consider a social group of n actors who will accomplish a certain goal or reach an agreement. Every actor in the group is represented by an element of the node set $V = \{1, \ldots, n\}$ of a network $G = (V, E)$, in which links (edges) $E \subseteq \{V \times V\}$ represent the relationships (friendship, any form of communication) among the actors. The set of neighbours of the actor i is denoted by $N_i = \{\, j \in V \,:\, (i, j) \in E\}$. Let $\mathbf{A} = [a_{ij}] \in \mathbb{R}^{n \times n}$ and $\mathbf{L}(G) = [l_{ij}] \in \mathbb{R}^{n \times n}$ be the adjacency matrix and Laplacian matrix, respectively, associated with graph G. The Laplacian matrix is defined as in 1.2.14 on page 30.

According to the consensus model (1.5.3), the information states of the actors evolve according to the single-integrator dynamics given by $\dot{x}_i(t) = u_i$, $i = 1, \ldots, n$, where $x_i(t)$ is the information state at time t, u_i is the information control input, and $x_i(0) = z_i$ is the initial state for actor i. We consider initial states of all actors to be assigned at random.

The continuous time consensus model is given by

$$x_i = \sum_{j \in N_i} a_{ij} \left(x_j(t) - x_i(t) \right), \;\; i = 1, \ldots, n, \tag{3.1.1}$$

where a_{ij} is the (i, j) entry of the adjacency matrix $\mathbf{A}$. The information state of each actor is driven toward those of her neighbours. Equation (3.1.1) describes the collective dynamics of the social group. Then, we can model consensus in a social

group by encoding the state of each individual at a given time t in a vector $\mathbf{x}(t)$. The group reaches consensus at $t \to \infty$ when $|x_i(t) - x_j(t)| \to 0$ for every pair of individuals, and the collective dynamics of the system is modelled in matrix form as $\frac{d\mathbf{x}(t)}{dt} = -\mathbf{L}\mathbf{x}(t),\ \mathbf{x}(0) = \mathbf{x}_0$ (see 1.5.3 on page 49), where $\mathbf{x}(t) = [x_1, \ldots, x_n]^T$ is the vector of the states of the actors in the system at time t, and $\mathbf{L}$ is the linear Laplacian operator previously defined, which captures the topology of the social network.

For the system represented by its associated network, the interaction among actors is considered to occur at a discrete time, thus the information state is updated using the discrete time consensus model given by

$$x_i(t+1) = x_i(t) + \epsilon \sum_i a_{ij} \left(x_j(t) - x_i(t)\right) \ \ i = 1, \ldots, n, \tag{3.1.2}$$

where a_{ij} is as before and ϵ is the time step (see 1.5.11 on page 52). The information state of each actor is updated as the weighted average of her current state and those of her neighbours. Equation (3.1.2) is then written in matrix form as $\mathbf{x}(t+1) = \mathbf{P}\mathbf{x}(t)$, and the matrix $\mathbf{P}$ is obtained as in 1.5.13 on page 53.

3.1.1.1 About the time for consensus

The second smallest eigenvalue of the Laplacian matrix, i.e., the graph connectivity μ_2, is known to have a main role in the convergence time of consensus models as it has been determined that the convergence speed of the average consensus is dictated by this eigenvalue [86], which also influences the performance and robustness properties of dynamical systems operating over a network.

Since we are interested in analyzing the influence of the algebraic connectivity on the time of consensus t_c, i.e., the time for which $|\vec{x_i} - \vec{x_j}| \leq \delta$, where δ is a given threshold, we state the following theorem:

Theorem 3.1. ***Theorem:*** *Let G be a connected graph with n nodes. Let $\langle t_c \rangle$ be the time of consensus averaged for all the nodes in the graph. Let μ_2 be the algebraic connectivity of G and $\vec{\phi}_2$ the corresponding Fiedler vector. Then*

$$\langle t_c \rangle \geq \frac{1}{n\mu_2} \sum_{p=1}^{n} \ln \left| \frac{\phi_2(p) \left(\vec{\phi}_2 \cdot \vec{x}_0 \right)}{\delta} \right|.$$

Proof. First, we write the spectral consensus equation for a given node p as

$$\vec{x}_t(p) = \sum_{q=1}^{n} \vec{x}_0(q) \sum_{j=1}^{n} \vec{\phi}_j(p) \vec{\phi}_j(q) e^{-t\mu_j}, \tag{3.1.3}$$

which represents the evolution of the state of the corresponding node as time evolves. Now, let us consider that the time tends to the time of consensus $t \to t_c$, where t_c is the time at which $x_t \to \left(\vec{\phi}_1^T \vec{x}_0 \right) \vec{\phi}_1$. Let us designate this time by t_c^-

$$\vec{x}_{t_c^-}(p) = \frac{1}{n} \sum_{q=1}^{n} \vec{x}_0(q) + \sum_{j=2}^{n} \left(\vec{\phi}_j(p) e^{-t_c^-(p)\mu_j} \sum_{q=1}^{n} \vec{\phi}_j(q) \vec{x}_0(q) \right), \tag{3.1.4}$$

here $t_c^-(p)$ means the time at which the node p is close to reaching the consensus state. Let $\langle \vec{x}_0 \rangle = \frac{1}{n} \sum_{q=1}^{n} \vec{x}_0(q)$ and let us write 3.1.4 as follows

$$\vec{x}_{t_c^-}(p) - \langle \vec{x}_0 \rangle = \sum_{j=2}^{n} \left(\vec{\phi}_j(p) e^{-t_c^-(p)\mu_j} \sum_{q=1}^{n} \vec{\phi}_j(q) \vec{x}_0(q) \right). \tag{3.1.5}$$

Let us select a node p such that $\vec{\phi}_2(p)$ gives the maximum value of the product $\vec{\phi}_2(p) \left(\vec{\phi}_2 \cdot \vec{x}_0 \right)$.

Since μ_2 corresponding to $j = 2$ is the smallest eigenvalue in the sum on the right hand of the expression, this terms tends to 0 slower than the terms for the other values of j. This means that, if we choose a small enough value of δ, the values of t_c and thus t_c^- will be very large. Thus, we can ensure that the left side of the equation is small enough that $\sum_{j=3}^{n} \left(\vec{\phi}_j(p) e^{-t_c^-(p)\mu_j} (\vec{\phi}_j \cdot \vec{x_0}) \right) < 0$. This implies that

$$\left(\vec{x}_{t_c^-}(p) - \langle \vec{x}_0 \rangle \right) < \vec{\phi}_2(p) e^{-t_c^-(p)\mu_2} \left(\vec{\phi}_2 \cdot \vec{x}_0 \right). \tag{3.1.6}$$

Now, because $\left|\vec{x}_{t_c^-}(p) - \langle \vec{x}_0 \rangle\right| \geq \delta$ we have

$$\delta \leq \left|\vec{x}_{t_c^-}(p) - \langle \vec{x}_0 \rangle\right| < \left|\vec{\phi}_2(p)\, e^{-t_c^-(p)\mu_2} \left(\vec{\phi}_2 \cdot \vec{x}_0\right)\right|. \tag{3.1.7}$$

Then, the time at which the consensus is reached $t_c(p)$ is bounded by

$$t_c(p) \quad \geq t_c^-(p) \geq \tfrac{1}{\mu_2} \quad \ln \left|\frac{\vec{\phi}_2(p)\left(\vec{\phi}_2 \cdot \vec{x}_0\right)}{\delta}\right|. \tag{3.1.8}$$

Finally, the average time of consensus is bounded by

$$\langle t_c \rangle \geq \frac{1}{\mu_2 n} \sum_{p=1}^{n} \ln \left|\frac{\vec{\phi}_2(p)\left(\vec{\phi}_2 \cdot \vec{x}_0\right)}{\delta}\right|, \tag{3.1.9}$$

which proves the result. □

If we are using a discrete-time approach like the one given in 3.1.2 then

$$\langle t_c \rangle \quad \geq \quad \frac{1}{\epsilon \mu_2 n} \sum_{p=1}^{n} \ln \left|\frac{\vec{\phi}_2(p)\left(\vec{\phi}_2 \cdot \vec{x}_0\right)}{\delta}\right|. \tag{3.1.10}$$

This last expression allows to analytically bound *a priori* the average time for consensus for a given network, provided the condition $\mu_2 < \mu_3$ holds, based on the knowledge of a few parameters which have been shown to be structurally and dynamically important for analyzing consensus processes [86, 136–139], i.e. the eigenvalue μ_2 and its corresponding associated eigenvector ϕ_2 of the Laplacian matrix $\mathbf{L}(G)$.

From the last results, we have two lower bounds for the times for consensus: expression 3.1.8 gives a bound for the highest time for consensus given by the time when the state of the last node of the network reaches the consensus value of the

system, while expression 3.1.9 provides a lower bound for the average time for consensus, which is given by the average of all the times of the nodes at consensus.

For illustration purposes, we calculated the average time for consensus for 12 real-world networks and found the values of the two bounds for these times. The first bound (t_{avgB}) is calculated following 3.1.9, and the second bound (t_{highB}) with expression 3.1.8,were we selected the node p as the one giving the maximum of the product $\vec{\phi}_2(p)\left(\vec{\phi}_2 \cdot \vec{x}_0\right)$.

These values are reported in table 3.1.1. The parameters for these calculations were $\delta = 10^{-4}$ with random vector of initial states $\vec{x}_0$ for each simulation. The values presented are the average of the results of 30 repetitions for each network.

Name	nodes	Type	*time*	t_{avgB}	t_{highB}	$\frac{t_{highB}}{t_{avgB}}$	$\frac{time}{t_{avgB}}$	$\frac{time}{t_{highB}}$
Benguela-A	29	eco	6.10	4.52	5.01	1.11	1.35	1.22
StMarks-A	48	eco	3.77	2.77	3.38	1.22	1.36	1.12
PRISON-SymA	67	soc	21.04	15.73	16.70	1.06	1.34	1.26
BridgeBrook-A	75	eco	11.80	8.66	8.72	1.01	1.36	1.35
Ythan1-A	134	eco	6.88	3.42	5.94	1.73	2.01	1.16
ElVerde-A	156	eco	8.01	3.79	6.39	1.68	2.11	1.25
LittleRock-A	181	eco	5.98	1.35	5.13	3.80	4.42	1.16
GD-mainA	249	inf	43.30	24.56	37.22	1.52	1.76	1.16
neurons-A	280	bio	6.26	2.72	5.54	2.03	2.30	1.13
Transc-yeast-mainA	662	bio	209.86	120.65	150.46	1.25	1.74	1.39
Software-VTK-main-sA	771	tec	70.05	38.36	43.86	1.14	1.83	1.60
Roget-mainA	994	inf	23.14	4.28	7.62	1.78	5.41	3.04

Table 3.1.1: Values of time for consensus and bounds for 12 real-world networks.

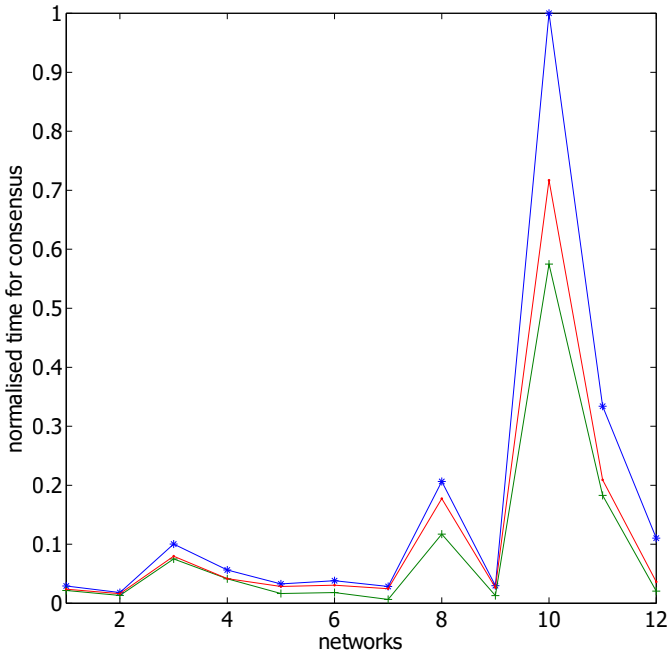


Figure 3.1.1: Normalised times for consensus and bounds for 12 real-world networks. Blue line: actual time for consensus. Green line: t_{avgB}. Red line: t_{highB}.

In figure 3.1.1 we can see that both bounds are always below the actual times for consensus, and that the second bound is tighter than the first one. The values of times were normalised to the highest value.

In figure 3.1.2 we have plotted the normalised times for consensus given in figure 3.1.1, and we have added the ratios of the actual time to every bound for comparison, confirming that the values of t_{highB} are tighter than the values of t_{avgB}, for the 12 networks analysed. The gap between the values of the ratios show the actual gap between the bound and the actual time for consensus, this means that the closer the value of the ratio is to 1, the better the bound is for that case, thus, in general, the bound given by 3.1.8 appears to be better for the network analysed.

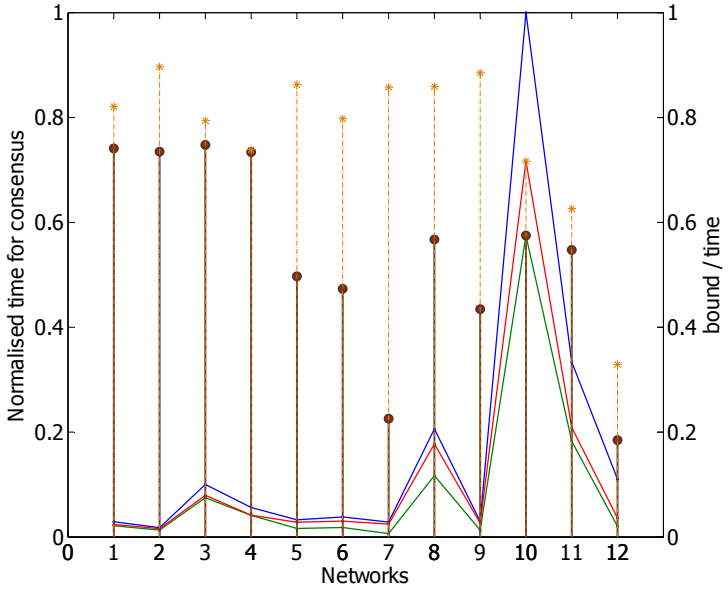


Figure 3.1.2: Normalised times for consensus and bounds for 12 real-world networks: blue line) actual time for consensus, green line) t_{avgB}, red line) t_{highB}. Ratios are given by stem plots: red filled circle) $\frac{t_{avgB}}{time}$, orange star $\frac{t_{highB}}{time}$

3.1.1.2 Consensus with leaders

Consensus is known to be influenced by a small group of leaders who guides the behaviour of the whole network [140–147]. The role of these drivers in the system controllability, and in particular their position in the network, has received great importance recently [14, 103, 111, 118, 122, 124, 148, 149]. In this situation a group of leaders indicates and/or initiates the route to the consensus, and the rest of the group readily follows their attitudes.

To take into account the influence of nodes acting as drivers we start from model (3.1.2). We consider that there exist one or multiple leaders who guide the entire group to the consensus through the effect produced by the rest of the group, which follows them [30]. In a leader–follower structure with a single leader, actors attempt to reach an agreement that is biased to the state of the leader, whereas in the case of multiple (stationary) leaders, all followers converge to the convex hull formed by the leaders' states (see 1.6.16 on page 58). We consider that leaders are

stationary, which means that their opinion is available for the other actors but is not modified during the process.

The set of all actors is divided into two subgroups: leaders and followers and, as a result, the vector of the states of all actors can also be divided into two parts: the states of leaders, $\mathbf{x}_l$, and the states of followers, $\mathbf{x}_f$. For a system with multiple stationary leaders, all the nodes can be labeled such that the first n_f represents the followers and the remaining n_l represent the leaders. The total number of actors in the system is then $n = n_f + n_l$, such that the Laplacian matrix associated with the social network G is partitioned as $\mathbf{L}(G) = \begin{bmatrix} \mathbf{L}_f & \mathbf{l}_{fl} \\ \mathbf{l}_{fl} & \mathbf{L}_l \end{bmatrix}$, where $L_f \in \mathbb{R}^{n_f \times n_f}$, $L_l \in \mathbb{R}^{n_l \times n_l}$, and $l_{fl} \in R^{n_f \times n_l}$ (see 1.6.6 on page 56).

We consider here that the leaders are stationary, thus their dynamics are given by $\dot{x}_l(t) = 0$, $i = n_f + 1, \ldots, n$, and the dynamics of the system is then $\begin{bmatrix} \dot{\mathbf{x}}_f \\ \dot{\mathbf{x}}_l \end{bmatrix} = -\mathbf{L}_p\mathbf{x} = -\begin{bmatrix} \mathbf{L}_f & \mathbf{l}_{fl} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{x}_f \\ \mathbf{x}_l \end{bmatrix}$ (see 1.6.9 on page 57), which can be expressed in its discrete time version as $\mathbf{x}(t+1) = (\mathbf{I}_n - \epsilon\mathbf{L}_p)\mathbf{x}(t)$, with $x(t) = [x_1(t), \ldots, x_n(t)]^T$, $\mathbf{I}_n$ is the identity matrix of size $n \times n$, and $\mathbf{L}_p$ is the Laplacian matrix of the network G, with each entry of the jth row equal to zero for $j = n_f + 1, \ldots, n$.

3.1.2 Modelling peer pressure

The consensus dynamic modelling assumes that the actors only interact with their directly connected neighbours to cooperatively achieve an agreement in the system [13]. However, in many real-world situations, the actors are exposed not only to their closest contacts but also to individuals who are socio-culturally close to them despite not being directly connected. For instance, this situation appears in actors' attitudes toward copying others. The predisposition of an actor to copy a behaviour depends not only on her friends' adoption of such behaviour but also on other, socio-culturally close people having a positive predisposition to that behaviour. For instance, adolescents adopt "binge drinking" not only by copying their mates but also by observing similar behaviour among others of a similar age, education, and social class. Then, we argue that this socio-cultural distance

can be captured in a model by considering the shortest path distance between two actors in their social group, being such shortest path distance the number of steps in the shortest path connecting the two actors. The influence that an actor receives/produces from/for others in her social network, i.e., peer pressure, decays as a function of this socio-cultural distance, which separates the two actors [21].

Peer pressure can then be modelled by considering the generalized Laplacian matrix which is based on the path matrices that characterize the existence of shortest paths between pairs of nodes in a graph [39]. These matrices were motivated by the problem of determining whether every node v_i of a graph can be visited by means of a process consisting of hopping from one node to another separated at distance (d) k from it. To construct this kind of Laplacian matrix, we need to take into account some definitions.

Definition 3.2. A k-hopping walk of length l is any sequence of (not necessarily different) nodes $v_1, v_2, \ldots, v_{l,} v_{l+1}$ such that $d_{i,i+1} = k$ for each $i = 1, 2, \ldots, l$. This k-hopping walk is referred to as a k-hopping walk from v_l to v_{l+1}.

This generalizes the concept of walk because a walk of length l is a 1-hopping walk of length l.

Definition 3.3. A k-hopping connected component in a graph $G = (V, E)$ is a subgraph $G' = (V', E')$, $V' \subseteq V$, $E' \subseteq E$, such that there is at least one k-hopping walk that visit every node $v_i \in V'$.

In order to solve the problem, the author generalizes the combinatorial Laplacian matrix of a graph, and for this purpose, he stated the next definition

Definition 3.4. The k-path degree $\delta_k(v_i)$ ($k \leq d_{max}$) of a node v_i is the number of irreducible shortest-paths of length k having v_i as an endpoint (see figure as example).

We have to note that $\delta_1(v_i)$ corresponds to the classical node degree, which is the number of edges incident to v_i.

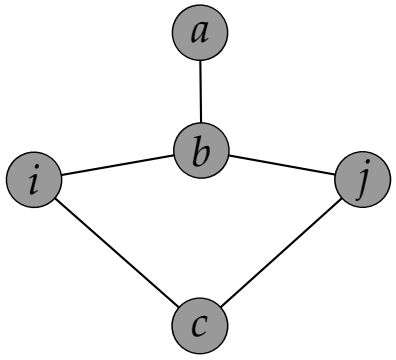


Figure 3.1.3: Illustration of the k-path degree of a node. Suppose we need to determine the k-path degree of length 2 for the node j. The total number of shortest paths that end in node j are three (one from node a and two from node i), but two of them start from node i, which are considered redundant. Thus, the number of non-redundant shortest paths ending in node j are two, so the k-path degree of length 2 for node j is then 2.

Then, the generalized Laplacian matrix is defined as follows:

Definition 3.5. The k-Laplacian matrix $\mathbf{L}_k$ ($k \leq d_{max}$) of a connected undirected graph $G = (V, E)$ is defined as the square symmetric $n \times n$ matrix whose entries are given by

$$L_k(i,j) = \begin{cases} -1 & d_{ij} = k, \\ \delta_k(i) & i = j, \\ 0 & otherwise. \end{cases} \tag{3.1.11}$$

Here, $\mathbf{L}_1 = \mathbf{L}$ is the combinatorial Laplacian matrix of a graph defined as in (1.2.14). The k-path Laplacian matrices are positive semidefinite, and as for the case of the combinatorial Laplacian, the path Laplacian matrices are singular M-matrices.

Consequently, the consensus dynamics model of equation (3.1.2) can be written as

$$x(t+1) = \left(\mathbf{I}_n - \epsilon \left(\sum \Delta_d \mathbf{L}_d\right)\right) \mathbf{x}(t), \tag{3.1.12}$$

where $\sum_d \Delta_d \mathbf{L}_d$ involves the d-Laplacian matrices and the coefficients indicate the strength of the interactions at distance $d \leq d_{max}(G)$, with $d_{max}(G)$ being the

maximum distance between any two nodes, or the diameter, of graph *the* G. Then for our case, d-Laplacian matrix is defined as

$$L_d(i,j) = \begin{cases} -1 & d_{ij} = d \\ \nu_d & i = j \\ 0 & otherwise \end{cases}, \tag{3.1.13}$$

where the expression $\nu_d(i)$ is the d-path degree of node i, i.e. the number of non-redundant shortest paths of length d having i as an endpoint.

The coefficients Δ_d should account for the decay in peer pressure for the socio-cultural distance between the actors of $\Delta_d \sim f(d)^{-1}$, where $f(d)$ represents a function of the distance d. These coefficients are expected to give more weight to shorter interactions than to the longer ones, so the influence of an actor over another decays with the separation among them. For our analysis we consider three different decay laws: [21, 39]

- Physical influences. This kind of interactions are present in situations where the communication among actors in a system displays a sort of spatial decay, this means that the effectiveness of transmission of information is better for two actors that are spatially closer. An example of this kind of situation is sensor systems, where these devices display low signal-to-noise ratio due to the spatial decay of the signal energy [150], another example can be found during earthquakes, where the aftershocks follow a spatial decay [151]. The interconnectivity between some neurons in mammalian neocortex decays exponentially with the inter-somatic distance [152]. Thus, we can consider two forms of decay where the strength of interactions would fade as the distance d between two actors grows.

-Power-law: $\Delta_d = d^{-\alpha}$

-Exponential: $\Delta_d = e^{-\beta d}$

- Social interactions. This kind of interactions, unlike the previous ones, would account for the benefits of a link between two actors in a network. This decay

was proposed in [21] as a way to account for the social contacts among nodes when a long-range interaction is present. The idea behind this factor is that if two individuals influence each other, then they have larger chance of becoming friends than two others which have null mutual influence. The new tie is considered to be created as an "investment" for the future, as an analogy of the future value of a growing annuity given in quantitative finance [153], so the mutual influence between two nodes separated at certain distance d is given by the future value of the investment that a new link created represent to them.

-Social: $\Delta_d = d\delta^{d-1}$

The last expressions account for different situations where the influence between two actors decays as a function of the distance between them, and the parameters α, β, and δ can be adjusted to consider the different strengths of peer pressure in the social network.

3.2 Leadership under peer pressure

The study of leadership in social groups has always intrigued researchers in the social and behavioural sciences [25, 140–143, 149]. Specifically, the way in which leaders appear in social groups is not well understood [144]. Leaders may arise either randomly in response to particular historical circumstances or from the individual having the most prominent position (centrality) in the social network at any time.

The centrality of an actor in her social group can be considered in several ways. The concept of actor centrality is related to the question "Which are the most *important* or *central* nodes in a network?" We study five centrality measures as criteria for potential leaders to come up from among the actors in a social group. The centrality measures considered are as follows: [5]

- *Degree centrality* (DC): This measure is considered the simplest in a network defined as the number of edges connected to a node. It has been used assuming that nodes with connections to many other nodes might have more

influence or access to information than those with few connections.The degree of a node can be expressed in a matrix as

$$\kappa_i = (\mathbf{A1})_i, \tag{3.2.1}$$

where $\mathbf{1}$ is a column vector of ones and $\mathbf{A}$ is the adjacency matrix of the network.

- *Eigenvector centrality* (EC): This measure appears as an extension of the degree of centrality. Eigenvector centrality is based on the question that not all neighbours are equivalent because, in some cases, the importance of a node is related to (and increased by) its neighbours, which may themselves be important. Thus, instead of giving only one point for each neighbour, this measure gives each node a score proportional to the sum of its neighbour's scores. The eigenvector centrality of node i is given by the ith entry of the principal eigenvector of the adjacency matrix

$$\varphi_i = \left(\frac{1}{\lambda_1} A\varphi_1\right)_i. \tag{3.2.2}$$

- *Closeness centrality* (CC): This index measures the inverse of the mean distance from a node to other nodes and characterizes the nodes according to their distance to all other nodes in the network. The closeness is defined as

$$CC(v) = \frac{n-1}{s(v)}, \tag{3.2.3}$$

where $s(v) = \sum_{w \in V(G)} d(w, v)$ is the distance sum of node v as defined in (1.1.10).

- *Betweenness centrality* (BC): This concept measures the extent to which a node lies on paths between other nodes. The nodes with high betweenness centrality may have considerable influence within a network because of their control over information passing through them.The index can be defined as

$$BC = \sum_i \sum_j \frac{\rho(i,k,j)}{\rho(i,j)},\ i \neq j \neq k, \tag{3.2.4}$$

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

- *Subgraph centrality* (SC): This measure is based on the notion that the importance of a node can be characterized by considering its participation in all closed walks for which it is the starting point. Subgraph centrality has been defined as

$$EE(i) = \left(\sum_{i=0}^{\infty} \frac{\mathbf{A}^l}{l!} \right)_{ii} = \left(e^{\mathbf{A}} \right)_{ii}. \tag{3.2.5}$$

To capture the influence of PP over the arising of leaders in social groups, we consider that the pressure that an individual p receives from q deteriorates proportionally with the social distance between p and q. The social distance is captured by the number of links in the shortest path connecting p and q. Mathematically, we model the mobilizing power between two individuals at distance d as $\Delta_d \sim f(d)^{-1}$, where $f(d)$ represents a function of the social distance (see equations (3.1.12) and (3.1.13)). The collective dynamics of the network under peers' mobilizing effects is described by the following generalization of the consensus model

$$\frac{d\mathbf{x}(t)}{dt} = -\left(\sum_d \Delta_d \mathbf{L}_d \right) \mathbf{x}(t),\ \mathbf{x}(0) = \mathbf{x}_0, \tag{3.2.6}$$

where $\mathbf{L}_d$ captures the interactions between individuals separated by d links in their social network, $\Delta_d \sim \frac{1}{d^\alpha}$ where the parameter α accounts for the strength of the PP pulling an individual into the consensus.

We consider that leaders' opinions may differ from the group's average opinion. We call this difference the divergence ∇_L, which is represented by the circumradius of the regular polygon (for a two-dimensional case) that covers all opinions of the leaders (see fig. 3.2.1). If the concerned problem is multidimensional, that is, if we are considering more than two opinions in the system, then the divergence is the circumradius of a hypersphere that covers the opinion of all leaders. A value of $\nabla_L = 0$ indicates that the leaders do not have difference on their states and are equal to the average consensus of the system.

We applied our model to two simple undirected random graphs and 15 real social (simple undirected) networks described in the dataset table on page 164. We allowed for every actor to have two opinions that were considered independent from each other, enabling a two-dimensional decoupled consensus process. The initial states of the followers were randomly assigned for every process with values in the range (0, 1).

For visualization purposes we selected, either randomly or from among the most central actors, only six leaders in every network. Their initial positions were assigned to have divergences of 0.1 and 0.2 from the average value of consensus. We considered that consensus was reached when the difference between two consecutive measures of disagreement was less than or equal to the threshold 1e - 07. We simulated consensus processes with and without PP. The normalised values of every network are reported in the tables on page 171 (see also in the same page 171 an explanation for reading the entries of the tables and the procedure for obtaining the times for consensus). Every value is the average of 50 repetitions and we used Matlab© to code and run the simulations (the code can be consulted in appendix C on page 198).

To illustrate our findings, we present the results obtained from the simulations for the network of workers in a sawmill (Sawmill network). First, we simulated the consensus process with no divergence in leaders initial positions. We computed the times for consensus with no PP and with different values of peer pressure according to the decays given on page 93. We chose leaders both randomly and by centrality criteria. All values were normalised to the highest value (see table 3.2.1).

	Random selection							
	No PP	PL-decay		Exp-decay		Social		
		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
	1.00	0.22	0.15	0.88	0.72	0.49	0.20	0.06
	Centrality-based selection							
	No PP	PL-decay		Exp-decay		Social		
Centrality		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
BC	0.80	0.21	0.14	0.73	0.62	0.43	0.19	0.05
CC	0.84	0.21	0.14	0.77	0.65	0.45	0.19	0.05
DC	0.83	0.21	0.14	0.76	0.64	0.45	0.19	0.06
EC	0.96	0.23	0.15	0.87	0.73	0.50	0.20	0.06
SC	0.89	0.23	0.14	0.85	0.69	0.46	0.20	0.06

Table 3.2.1: Normalised consensus time for the Sawmill network with no divergence on leaders positions with respect the average consensus. All initial states of the leaders are equal to the average consensus.

Then we allowed a divergence on leaders positions with respect to the average consensus. This divergence ∇_L took the values of 0.1, 0.2 and 0.5, and the conditions for consensus (selection of leaders, peer pressure decay functions) were as stated before. The results for all the set of networks used for this analysis can be consulted in the correspondent appendix on page 171.

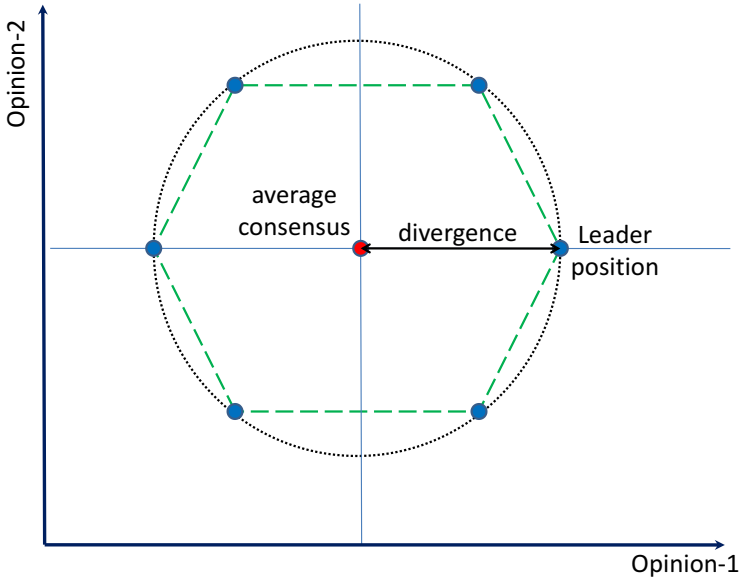


Figure 3.2.1: Leaders with divergence. Distribution of leaders positions (blue points) around the average consensus value of the system (red point), which represents the centroid of the convex hull spanned by the leaders.

3.2.1 Simulations results

We compare the hypotheses about the random arising of good leaders - those who significantly reduce the time for reaching consensus in a network - to those in which leaders come out from the most central individuals. In general, we observe that the leaders coming from the most central individuals are better in leading the consensus than those coming out randomly. However, when there is certain level of PP over the actors, the situation changes dramatically (see fig. 3.2.2). First, the time to reach consensus significantly decreases to less than 20% of the time needed when no PP exists. Second, a leader randomly selected in the network could be as good as one selected from the most central actors when PP exists in the system.

In roughly half of the 15 social networks studied (see tables on page 171) we observed the following pattern: randomly chosen leaders in these networks are

better in leading the consensus than those selected from the most central individuals. This is the case of the friendship network of injected drug users in figure 3.2.2c. The last situation appears when the network has the leaders distributed through diverse communities in the network.

A *community* is a group of individuals who are more tightly connected among themselves than with the other actors in the network [5]. Actors in one of these communities reach consensus among themselves easily, but it is difficult to reach consensus between different communities. Most central actors in such networks are frequently located in a single community. When they arise as leaders, they drive consensus only in their community but not in the global network. In contrast, when leaders arise randomly, they more likely come out simultaneously in different communities, a situation that favours global agreement in the network.

In order to corroborate the last observation, we constructed a random network with communities as illustrated in figure 3.2.3. In this network we were able to control the number of communities as well as the connectivity within and among each community. These random network with communities comprised a simple, undirected random graph with 500 nodes and 10 communities. For constructing this kind of graph we used the C program for generating benchmarks for community detection given in [154].

In our simulations with the network with 10 communities, we first allowed 10 leaders to be selected randomly, and then by their global highest centrality values. We recorded the average time for consensus of 50 repetitions. Next, we allowed the existence of 10 community-based leaders, which means that this time leaders were chosen by their local highest centrality values. This procedure lead to having one leader in each community, which corresponded to the one with the highest centrality in the community. The results of the simulations are shown in tables 3.2.2 and 3.2.3.

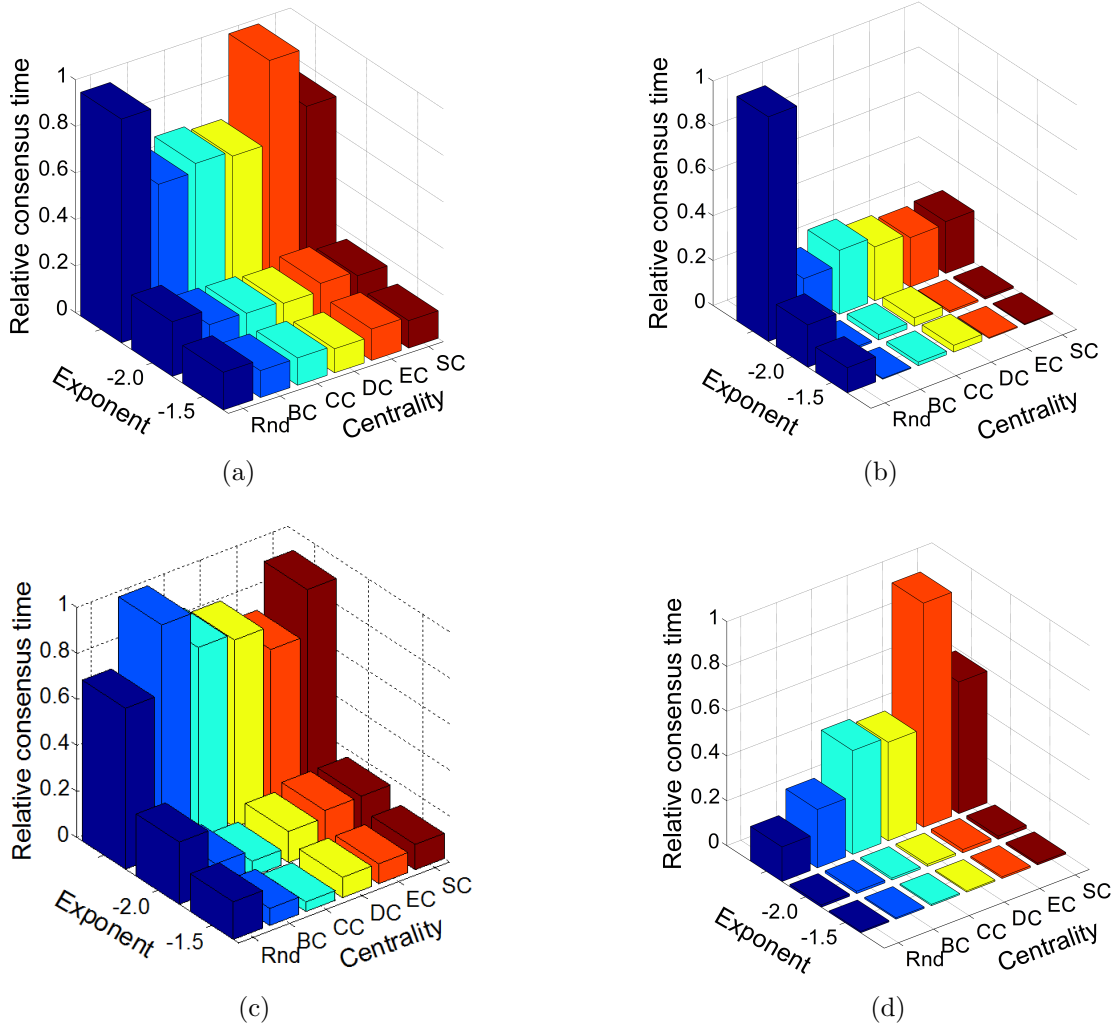


Figure 3.2.2: Random and centrality-based selection of leaders. The performance of nodes leading consensus is analysed according to randomness (Rnd), betweenness (BC), closeness (CC), degree (DC), eigenvector (EC), and subgraph (SC) centrality. The peer pressure is modelled by $\Delta_d \sim \frac{1}{d^\alpha}$,with α equal to 1.5 and 2.0. The third line corresponds to no peer pressure. (a) Communication network among workers in a sawmill. (b) Elite corporate directors. (c) Friendship network of injected drug users in Colorado Springs. (d) Random network having communities.

	Random selection							
	No PP	PL-decay		Exp-decay		Social		
		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
	0.15	0.01	0.01	0.12	0.06	0.04	0.01	0.003
	Centrality-based selection							
	No PP	PL-decay		Exp-decay		Social		
Centrality		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
BC	0.26	0.01	0.01	0.18	0.09	0.06	0.01	0.003
CC	0.46	0.01	0.01	0.32	0.16	0.10	0.02	0.002
DC	0.44	0.01	0.01	0.32	0.16	0.10	0.02	0.003
EC	1.00	0.01	0.01	0.62	0.27	0.16	0.03	0.003
SC	0.59	0.01	0.01	0.34	0.15	0.09	0.02	0.004

Table 3.2.2: Normalised consensus times for a random graph with 10 communities (10 leaders) with and without PP. Leaders were selected according to their global centrality values.

	Random selection							
	No PP	PL-decay		Exp-decay		Social		
		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
	1.00	0.06	0.04	0.77	0.41	0.27	0.08	0.02
	Centrality-based selection							
	No PP	PL-decay		Exp-decay		Social		
Centrality		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
BC	0.54	0.07	0.04	0.42	0.28	0.17	0.06	0.02
CC	0.40	0.07	0.04	0.31	0.22	0.14	0.06	0.02
DC	0.30	0.07	0.05	0.26	0.20	0.13	0.07	0.02
EC	0.43	0.07	0.04	0.34	0.25	0.15	0.06	0.02
SC	0.29	0.07	0.04	0.25	0.19	0.12	0.06	0.02

Table 3.2.3: Normalised consensus times for a random graph with 10 communities (10 leaders) with and without PP. Leaders were selected according to their local centrality values (by community).

When the leaders were selected from the global highest centrality, we observed the previously described pattern of leaders being selected by random: they better lead the consensus than those coming from the most central individuals (see fig. 3.2.2d). This situation was modified when the leaders were selected from the community-

based centrality. In this case, the leaders coming from the most central local nodes were significantly better at reaching consensus than those coming out randomly.

These results suggest the necessity of considering community leaders in social networks as effective mobilizers of actors throughout the network. We have observed that the leaders arising on the basis of their community positions exhibit greater success in reaching consensus than those randomly arising in the network. However, when appropriate PP exists, leaders who effectively reach consensus arise regardless of their position in their communities.

The leaders in a social group do not always exhibit a high level of cohesiveness. We posit that the leaders' capacity to lead the consensus in a network depends on their divergence of opinions. A cohesive group of leaders can more effectively lead the social group than leaders with larger divergences among their opinions. To model leader cohesiveness we used the divergence parameter ∇_L previously defined on page 96.

The effects of the divergence in the system dynamics, can be seen on the simulation results for the social network Sawmill with different PP intensities. We increased the value of divergence on leaders' opinions from zero to a maximum of 0.5 and computed the consensus times for all divergence values. The complete results are reported in the corresponding tables in appendix B on page 171. For the case of randomly selected leaders, times increased along with divergence from 13.18% to 80.24% (see table 3.2.4). We highlight that PP influenced the trajectories of followers' opinions, precisely directing them toward the consensus space. At the consensus point, the final positions were more cohesive, indicating more homogeneous final opinions in the system (see fig. 3.2.5).

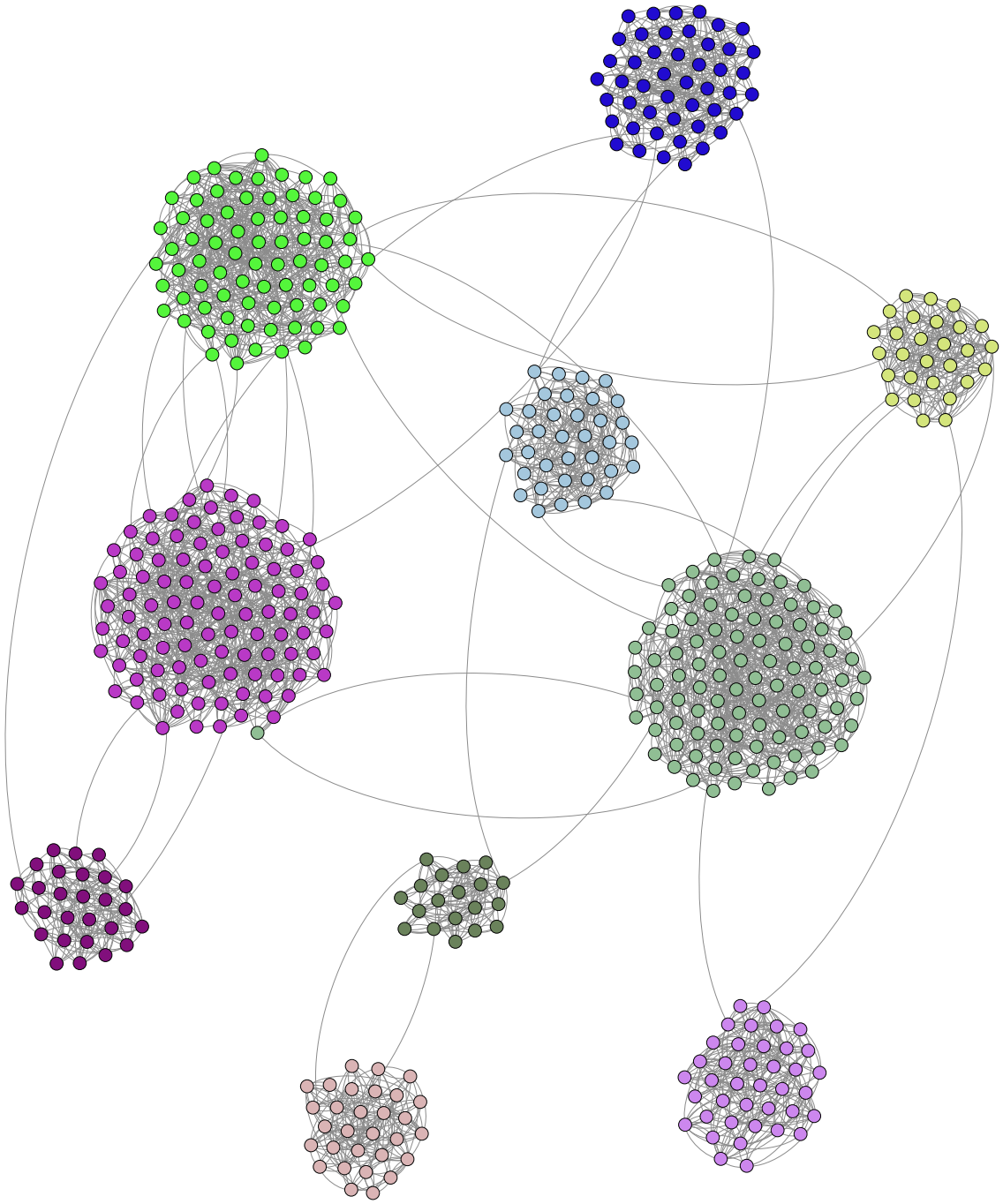


Figure 3.2.3: Random network with communities generated with the benchmarks for testing community detection algorithms. The parameters were set to get 10 well defined communities in a network of 500 nodes.

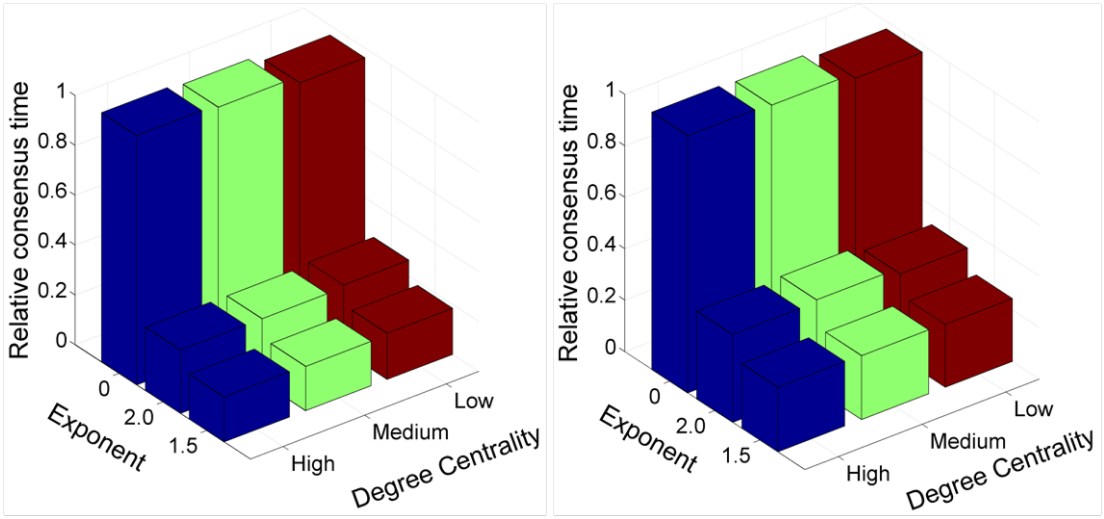
Divergence	Average time for consensus	% Increase in time for consensus
0	1,026.80	-
0.1	1,162.18	13.18
0.2	1,372.40	33.65
0.5	1,850.72	80.24

Table 3.2.4: Comparison of time for consensus for the Sawmill network with different values of divergence. The effect of the variation on divergence is reflected in the percentage of increase in time.

To examine the influence of the leaders' cohesiveness on consensus we use the results for the friendship network of workers in the sawmill (see fig. 3.2.5) with either no PP (left plots) or with PP modelled by $\Delta_d \sim \frac{1}{d^2}$ (right plots). The values of leader divergence range from 0.0 to 0.2. The lack of leader cohesiveness significantly increases the time to consensus when there is no PP as we can see in table 3.2.4. In addition, the cohesiveness of the group, measured by the standard deviation at consensus ∇_G, is very poor for large values of ∇_L (∇_G=154.6, 183.6, and 226.9 for ∇_L=0.0, 0.1, and 0.2, respectively), which indicates highly heterogeneous group opinions. However, when PP exists, the situation dramatically changes. First, the time to consensus does not increase as drastically with the decrease of leader cohesiveness. Second, group cohesiveness at the consensus is very high even for the lowest leader cohesiveness (∇_G=27.0, 35.4, and 33.0, for ∇_L=0.0, 0.1, and 0.2, respectively). In short, when PP is absent, leader cohesiveness plays a fundamental role in the time needed to reach consensus and in group cohesiveness at the consensus. When PP is present, the time needed to reach consensus and group cohesiveness are largely independent of the degree of divergence in the leaders' opinions, and the consensus is driven primarily by the influence of the nearest neighbours and PP.

Due to the recent results about the role of low-degree nodes in controlling complex networks [14] we also tested the role of PP over these potential drivers. Our results show again that nodes can be good leaders regardless of their centrality in the network when PP exists in the system as shown in figure 3.2.6, and in contrast with previous results [14, 118, 124]. In other words, under the appropriate PP any individual in a social group could be a good leader independently of her position in the network. This result adds a new dimension to the problem of network

controllability [14, 111, 122, 148] by demonstrating that PP is a major driving force in determining that potential controllers can appear in the network independently of their centrality (see fig. 3.2.4) and degree distribution (see fig. 3.2.6).



(a) (b)

Figure 3.2.4: Influence of PP on selection of leaders among the nodes with high, medium and low degree centrality. (a) The HighTech network and (b) the network of social dating (social3).

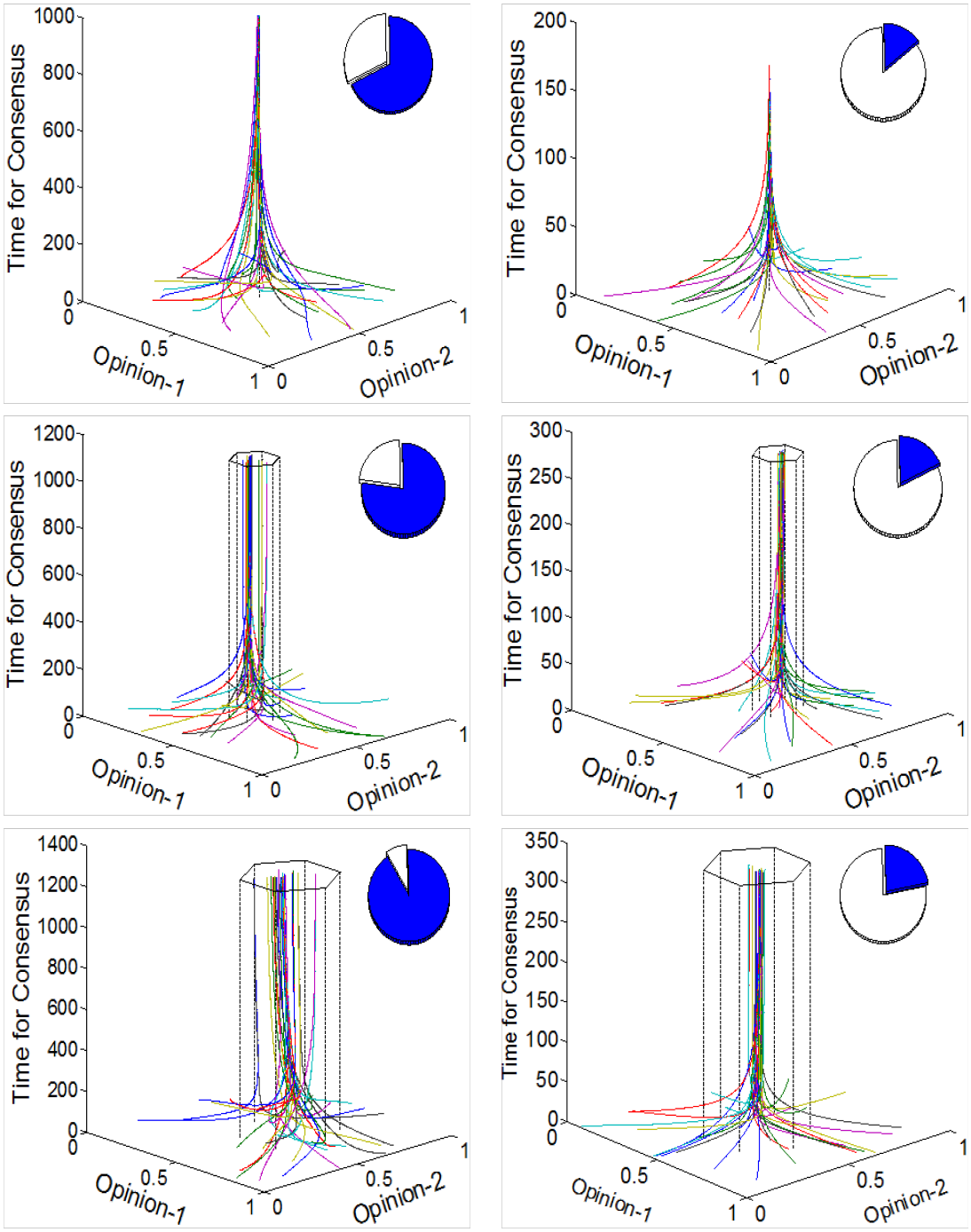


Figure 3.2.5: Leaders' cohesiveness and consensus. Analysis of the influence of leaders cohesiveness on the time to reach consensus in the communication network among workers in a sawmill without (left plots) and with (right plots) PP. The leaders' divergences used in this figure are: 0.0 (top), 0.1 (middle), and 0.2 (bottom). The time to reach consensus (in blue) relative to a total time of 1,500 units is showed as insets.

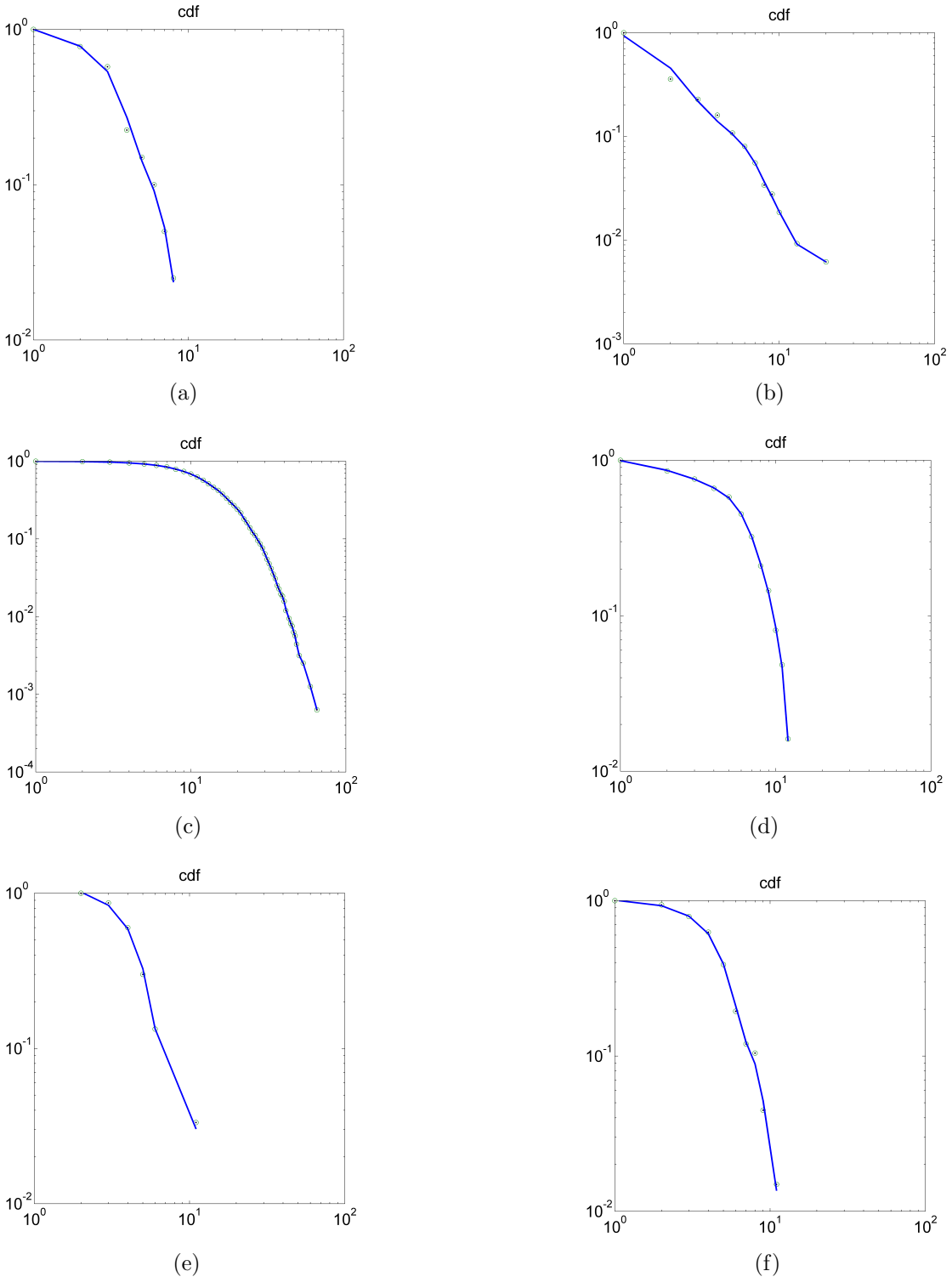


Figure 3.2.6: Cumulative degree distribution (cdf) of social networks. (a) BF23, (b) ColoSpg, (c) Corporate, (d) Dolphins, (e) MathMethod, and (f) Prison.

3.3 Diffusion of innovations under peer pressure

Another area that has received great research attention is the diffusion of innovations [155–158]. The essence of the diffusion process is the information exchange by which one individual communicates a new idea to one or several others; thus, diffusion is a special type of communication concerned with the spread of messages perceived as new. In its most elementary form, the main elements in the diffusion process are: [155]

- An innovation (message or information);
- Communication channels, through which messages are conveyed from one individual to another;
- Time of diffusion; and
- The social system through which the process occurs - a set of interrelated units engaged in joint problem solving to accomplish a common goal.

The members or units of a social system may be individuals, informal groups, organizations, and/or subsystems. All members cooperate at least to the extent of seeking to solve a common problem. This common objective binds the system.

Most innovations have a sigmoid-shaped (S-shaped) rate of adoption. The slope of the curve varies with every innovation. Certain new ideas diffuse relatively rapidly, and its S-curve is quite steep. Other innovations may have a slower rate of adoption, reflected by a more gradual S-curve. This behavior indicates the rate of adoption, i.e., the number of adopters of the new idea throughout time. The behavior of the adopters builds the form of the innovation process as follows [155]: Initially, few individuals adopt the innovation in each period; these are the innovators. Soon, the diffusion curve begins climbing as an increasing number of individuals adopt. Then, the trajectory of the rate of adoptions begins to level

off, as fewer individuals remain who have not yet adopted. Finally, the S-curve reaches its asymptote and the diffusion process is complete.

Most individuals do not evaluate an innovation on the basis of scientific studies regarding its consequences; most people primarily depend on a subjective evaluation from individuals such as themselves who previously adopted the innovation, i.e., a dependency on the communicated experience of near peers.

In diffusion networks, certain individuals play different roles in a social system, and these roles affect diffusion. Certain members of the system function as opinion leaders: individuals who can influence others, and who are often identified and used to assure better diffusion of the information.

Here we consider the hypothesis that PP plays a fundamental role in innovation adoption or rejection. To test our hypothesis, we studied two datasets in which some diffusion of innovations process was followed for different periods of time.

- The network from the study Mathematical Method (MathMethod) [159]: This innovation concerns the diffusion of a new mathematics method in the late 1950s. It was instigated by top mathematicians and sponsored by the U.S. National Science Foundation and the U.S. Department of Education. The diffusion process was successful because most schools adopted the new method. The example traces the diffusion of the modern mathematical method among school systems that combine elementary and secondary programs in Allegheny County (Pennsylvania, U.S.). All school superintendents who were in office for at least two years were interviewed. Among other things, the superintendents were asked to indicate their friendship ties with other superintendents in the county through the following question: Among the chief school administrators in Allegheny County, who are your three best friends? The researcher analyzed the friendship choices among superintendents who adopted the method and who were in office for at least one year before the first adoption, indicating that they could have adopted earlier. Unfortunately, the researcher did not include the friendship choices by superintendents who did not receive any choices themselves. In our study, the network represents the friendship ties among the 30 superintendents who were part of the connected component, and the times for adoption represent

the year in which the adopter chose the new mathematical method: 1-1958, 2-1959, 3-1960, 4-1961, 5-1962, and 6-1963.

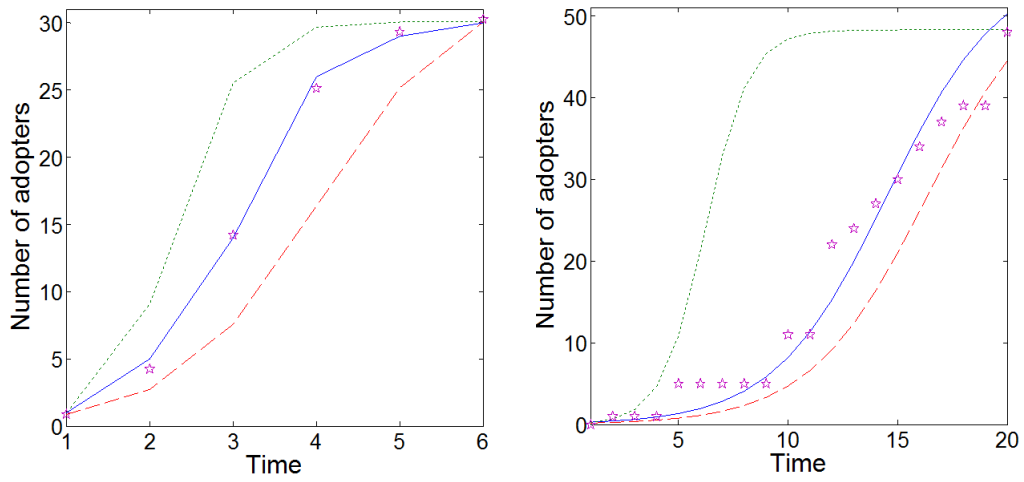
- Networks from the study Brazilian Farmers [160]: "Diffusion and Adoption of Innovations in Rural Societies, 1952–1973," was a longitudinal study on how Brazilian farmers (BF) adopted hybrid seed corns. The study was part of a broader, three-phase research project concerned with the spread of modern technology in Brazil, Nigeria, and India. The data files reflect the second phase, which examined personal factors influencing farmers' innovative agricultural behavior. Villages were selected from the total sample of Phase II villages. The groups of people were divided into different communities according to different variables, and the social networks of friends among the people in each community were retrieved. The dataset used for our study includes the social network of friendship ties and the cumulative number of adopters of the new technology over 20 years among the individuals in the giant connected component for three different communities of the study, identified as communities 23, 70, and 71 [161].

We applied our consensus model to these networks, and the average time of 50 repetitions was divided, into six intervals for the Mathematical Method, and into 20 intervals for the BF networks. We then counted the number of actors or nodes that had reached the value of average consensus at every time step by measuring the difference between every node's position and the average consensus. As a general rule, when the absolute value of this difference was less than or equal to a certain value ϖ (for out case, ϖ=0.04), we considered the node to be in agreement. This process was conducted with and without PP, and considering a power-law decay to simulate the strength of PP.

The cumulative average number of nodes in agreement at every interval is shown in tables 3.3.1-3.3.4, for every empirical network studied. We also have included the empirical cumulative number of adopters for every diffusion process. We varied the values of the parameter α to obtain behaviors that more effectively would follow the empirical patterns. We divided these values into two classes: moderate PP ($5.0 \leq \alpha \leq 6.0$) and high PP ($3.0 \leq \alpha \leq 4.0$).

From our results, we can see in figure 3.3.1 the number of actors that adopted the respective innovations at different times for 2 of the networks studied (the complete

set of figures is shown in appendix B on page 193). These values correspond to the number of adopters observed empirically in field studies. The simulations follow perfect sigmoid curves and we can observe that when there is no PP effect, the diffusion curves predict slower rates of adoption than those empirically observed. For example, the empirical evidence demonstrates that 50% of schools adopted the new math method in roughly three years, whereas the simulation without PP predicts a period of four years of a total of six years. In the case of the Brazilian farmers, the empirical time for 50% of the farmers to adopt the innovation is roughly 12 years, whereas the simulation without PP predicts 16 years of a total of 20 years. When the model uses strong PP, the diffusion curves display very rapid adoption rates, which are far from the reality of the empirical evidence in both cases. However, using a moderate PP predicts very well the outputs of the empirical results in both studies. These PP values are found by a reverse engineering method, but the important message is that a certain PP level is necessary to describe the empirical evidence on the diffusion of innovations in social groups. These results demonstrate that interpersonal communication alone cannot sufficiently explain the process of innovation adoption in a social group. The pressure exerted by the social group plays a fundamental role in shaping this important social phenomenon. Our model describes effectively PP's role in these and other important phenomena, consistent with our intuition and with the existing empirical evidence.



(a) (b)

Figure 3.3.1: Diffusion of innovations under PP. (a) Adopters of a new mathematical method among US colleges in a period of 6 years ($\alpha_{moderate} = 5,\ \alpha_{high} = 4$). (b) Adopters of the use of hybrid seed corns among Brazilian farmers (BF23) for a period of 20 years ($\alpha_{moderate} = 5.9,\ \alpha_{high} = 4$). Experimental values are given as stars and the simulation with no (broken red line), moderate (continuous blue line) and strong (dotted green line) PP are illustrated.

Periods	Adopters (empirical)	Avg. Adopters (simulation)		
		No PP	$\alpha = 4$	$\alpha = 5$
1	1	1.1	1.3	1.1
2	5	2.7	9.0	4.7
3	14	7.7	25.6	13.7
4	26	23.6	29.8	27.8
5	29	28.3	30	29.4
6	30	30	30	30

Table 3.3.1: Cumulative average nodes in agreement for the Mathematical Method network (MathMethod) with and without PP, and empirical values of the adopting process.

Periods	Adopters (empirical)	Avg. Adopters (simulation)		
		No PP	$\alpha = 4$	$\alpha = 5.9$
1	1	0.8	1.0	0.8
2	1	1.3	1.8	1.2
3	1	1.2	2.0	1.1
4	1	1.5	3.0	1.5
5	3	1.8	7.2	1.7
6	3	2.0	15.8	2.2
7	3	2.3	23.4	2.4
8	4	2.5	26.0	4.1
9	4	4.0	32.3	9.0
10	6	5.6	35.9	10.9
11	6	7.2	37.2	11.5
12	11	8.9	38.5	12.5
13	13	9.5	38.9	13.5
14	15	10.3	38.9	15.8
15	19	12.3	38.9	17.3
16	23	13.4	38.9	19.8
17	33	19.0	38.9	24.6
18	33	24.9	38.9	26.9
19	37	26.4	38.9	30.4
20	38	27.1	38.9	32.9

Table 3.3.2: Cumulative average nodes in agreement for the Brazilian Farmers, community 23 network (BF23) with and without PP, and empirical values of the adopting process.

Periods	Adopters (empirical)	Avg. Adopters (simulation)		
		No PP	$\alpha = 4$	$\alpha = 5.6$
1	0	0.9	0.8	0.7
2	0	1.7	3.2	1.8
3	0	1.8	9.5	3.0
4	0	3.3	14.3	4.5
5	14	4.9	23.4	5.6
6	14	4.7	36.2	8.6
7	15	5.8	45.4	13.7
8	18	6.4	47.7	16.5
9	20	6.6	47.8	18.9
10	28	8.3	47.8	21.0
11	29	10.2	47.8	23.5
12	31	11.8	47.8	25.9
13	31	14.0	47.8	28.4
14	36	16.5	47.8	30.7
15	38	19.1	47.8	33.1
16	42	22.7	47.8	37.8
17	44	27.3	47.8	42.1
18	44	31.7	47.8	45.6
19	45	35.4	47.8	47.4
20	46	39.0	47.8	47.5

Table 3.3.3: Cumulative average nodes in agreement for the Brazilian Farmers, community 70 network (BF70) with and without PP, and empirical values of the adopting process.

Periods	Adopters (empirical)	Avg. Adopters (simulation)		
		No PP	$\alpha = 4$	$\alpha = 6.6$
1	0	0.5	0.7	0.7
2	1	0.9	1.9	1.1
3	1	1.2	4.0	1.4
4	1	1.1	7.3	1.6
5	5	1.0	11.8	3.2
6	5	1.2	21.6	5.5
7	5	2.1	32.2	6.4
8	5	2.9	37.3	6.6
9	5	3.6	46.0	7.0
10	11	4.6	47.8	7.9
11	11	6.1	48.4	9.6
12	22	8.5	48.4	12.7
13	24	11.7	48.4	17.3
14	27	16.4	48.4	23.8
15	30	21.9	48.4	31.5
16	34	26.6	48.4	38.2
17	37	30.8	48.4	43.1
18	39	36.2	48.4	45.9
19	39	40.8	48.4	47.4
20	48	44.5	48.4	47.6

Table 3.3.4: Cumulative average nodes in agreement for the Brazilian Farmers, community 71 network (BF71) with and without PP, and empirical values of the adopting process.

Summary

In this chapter we proposed a way to explore influence of the combined action of direct and indirect peer pressure on the dynamics of social groups trying to reach consensus. We have extended a basic model of consensus by introducing a generalised Laplacian matrix which allows the actors to interact those peers that are not directly connected to them. We proposed the use of three different decay factors that can be tuned to adjust the strength of such long-range interactions. We applied our model to different social networks and confirmed that peer pressure is an important factor to take into account while studying consensus processes, leadership and diffusion of innovations. Some of the main results are that the leadership role of some actors in a social network can be faded with the presence of peer pressure. We also found that the adverse effect on reaching consensus due to the presence of communities in the structure of the network vanishes as the strength of peer pressure increases. Finally, by studying the diffusion of innovations processes we found that a certain amount of peer pressure is necessary to reproduce the empirical results coming from two studies on the field.

As a result of our theoretical analysis, we have derived mathematical expressions of two lower bounds for the consensus time for any network in terms of the second eigenvalue of the Laplacian matrix, which has been proved to dictate the rate of convergence of the consensus model. We analysed the case of 12 real-world networks for illustrating the behaviour of these bounds. The results point to the fact that the bound given by 3.1.8 can be the best, although a more extensive analysis must be carried out to confirm this.

Chapter 4

ON THE TIME FOR CONSENSUS AND THE STRUCTURE OF COMPLEX NETWORKS

Our empirical results from chapter 3 support the intuitive idea that the effectiveness of those nodes acting as drivers (leaders) on a consensus process depends largely on the structure of a network, i.e. if the structure presented communities, a good distribution of leaders among the communities would benefit the consensus. Here we explore correlations between the time for consensus and some general characteristics of networks. To enrich the study we expanded our dataset to take into account different kinds of real networks in addition to the social networks studied in the last chapter. We see that among all the characteristics, those related to the size and distance show stronger correlations with the time for consensus.

4.1 Time for consensus and some structural indices from networks.

We explore possible correlations between the time for consensus and different structural indices in complex networks. We divide these indices between two groups: the first group comprises the average node degree, density, average clustering coefficient and transitivity, while the second group includes the average path length, algebraic connectivity, diameter and average distance-sum. The reason of this division is twofold: first, we try to keep most degree-related and distance-related indices separated from each other, and second by dividing the complete set we can gain better visualisation of their behaviour. We briefly recall the formal definition of some of these concepts.

4.1.1 Structural characteristics.

We briefly remind the definitions for some structural characteristics of a network used for the present analysis.

- Average node degree: The average node degree of network is defined on page 20 as $\bar{k} = \frac{1}{n}\mathbf{1}^T\mathbf{k} = \frac{1}{n}\sum_{i=1}^{n} k_i$.
- Density: The density of a graph is defined as:

$$\varrho = \frac{2m}{n(n-1)} \tag{4.1.1}$$

where m is the number of edges.

- Average (Watts-Strogatz) clustering coefficient: From 1.1.13 on page 25 we have that the average clustering coefficient of a network is: $\bar{C} = \frac{1}{n}\sum_{i=1}^{n} C_i$, where C_i is the clustering coefficient of node i as defined in 1.1.12 on page 25.
- Transitivity: In 1.1.14 on page 26, transitivity is defined as $C = \frac{3|C_3|}{|P_2|}$, this metric relates the number of triangles C_3 with the number of connected triples P_2 (2-paths).

- Average path length. The definition of this metric comes from 1.1.11 on page 25 $\bar{d}(G) = \frac{1}{n(n-1)} \sum_{u,v \in V} d(u, v),\ d_{u,u} = 0$.
- Algebraic connectivity: on page 31 it is mentioned that the second smallest eigenvalue, $\mu_2(\boldsymbol{L})$ of the Laplacian matrix is commonly referred to as the algebraic connectivity of the network which has been related with dynamical properties of networks [41].
- Diameter. From 1.1.9 on page 24 we have that the diameter of a network is $diam(G) = \max_{x,y \in V(G)} \{d(x, y)\}$.
- Average distance-sum of a node. 1.1.10 on page 24 gives the definition of distance-sum of a node u as $s(u) = \sum_{v \in V(G)} d(u, v)$.

We take the arithmetic mean of the distance-sum values of all nodes in a network as a parameter of this characteristic for our analysis purposes, thus we have that

$$\bar{s}(u) = \frac{s(u)}{n} \tag{4.1.2}$$

.

4.1.2 Correlations between time for consensus and some structural characteristics.

We analyse the set of social networks used in chapter 3 plus other types of networks for enriching our results: biological (14), technological (12), informational (5), and ecological (17). This makes a total of 64 networks. We obtained the average time for consensus of 100 repetitions for all the dataset, keeping the parameter of 6 randomly selected leaders for each simulation. We used the statistical analysis software SPSS© to create the scatter plots in figures 4.1.1 and 4.1.2.

We can see in fig. 4.1.1 that among the structural characteristics considered, density is the one that seems to have stronger (negative) correlation with the time for consensus, indicating that the higher the density, the lower the time taken for reaching consensus in a system. We confirm this visual output by the

correspondent numerical result given in table 4.1.1 where we can see that the Pearson coefficient is -0.957. This not surprising result can be explained by the fact that a denser network implies more number of edges, which for diffusion processes means more channels for sharing information among nodes, and thus a better promotion of an agreement in the system. It is worth noticing that all the characteristics in this table show negative correlations with the time for consensus, and positive correlations among themselves. Regarding the relation with the clustering coefficient, we know that this metric is a measure of the number of mutual neighbours of adjacent nodes, which can affect the synchronisation of a network. Large values of clustering mean that there are many triangles, and this enhances consensus in a network.

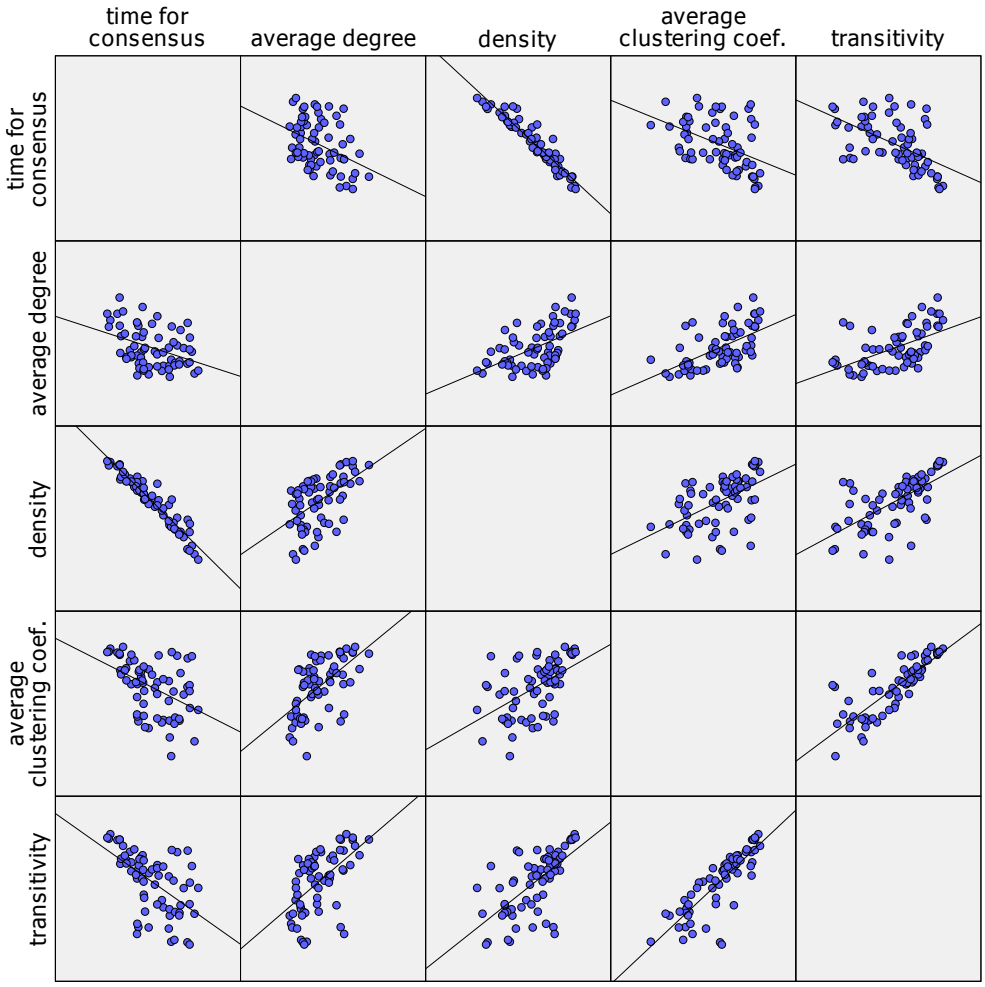


Figure 4.1.1: Log-log matrix scatter plots for time of consensus versus average node degree, density, average clustering coefficient and transitivity for 64 networks analysed. The black line shows the best linear fit for every case.

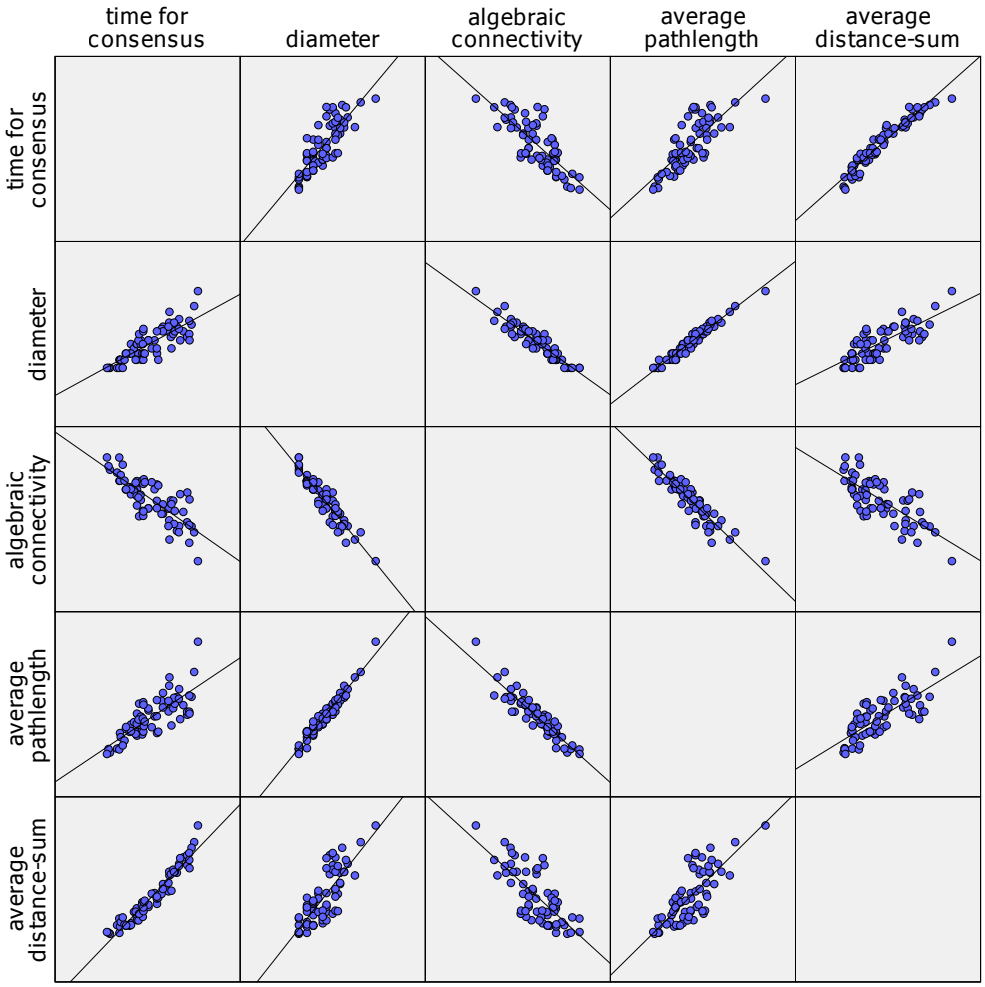


Figure 4.1.2: Log-log matrix scatter plots for time of consensus versus diameter, algebraic connectivity, average pathlength and average distance-sum for 64 networks analysed. The black line shows the best linear fit for every case.

As for the second group of structural characteristics, we can see stronger correlation values for all of them, where the average distance-sum is the one with higher (positive) correlation, as can be noticed from the output in table 4.1.2. The correlation coefficient between the time for consensus and the average distance-sum is 0.964. For the case of algebraic connectivity μ_2, we have that the correlation is negative, indicating that the higher the algebraic connectivity, the lower the time for consensus. Algebraic connectivity has been related with dynamic properties of networks, indicating how easily the network would synchronise, hence how easy a

network can reach consensus. We know that an upper bound for the value of the algebraic connectivity of a network of size n is given by [162]

$$\mu_2 \leq \frac{8k_{max}}{diam^2} log_2^2 n. \tag{4.1.3}$$

where k_{max} and $diam$ are respectively the maximum degree and the diameter of the network.

If we consider the above relation together with the analytical result given in **??** on page ?? for the time for consensus, then this confirms that the time for consensus is directly proportional to the diameter of a network in agreement with the result in table 4.1.2. It is also evident from 4.1.3 that the diameter provides an inverse measure of the vertex connectivity, this means that two nodes are weakly connected if their shortest connection is through many other nodes, i.e. if the diameter is large.

Like the diameter, the average path length among nodes is an inverse measure for the connectivity of the network, which can be noticed in the following bound [163]

$$\frac{1}{\mu_2} \leq \frac{(n-1)\bar{d}}{2} - \frac{n-2}{4}, \tag{4.1.4}$$

here we can see that networks with small number of nodes n and small average pathlength $\bar{d}$ show small inverse algebraic connectivity, which makes them present better properties for reaching consensus. Moreover, from the lower bound [163]

$$\left(\left\lfloor \frac{2\bar{d}(n-1) - n}{2n \ln(n-1)} \right\rfloor - \frac{1}{4} \right) \frac{4}{k_{max}} \leq \frac{1}{\mu_2}, \tag{4.1.5}$$

we see that if n is small and $\bar{d}$ is relatively large, the bound is also large and a consensus would be more difficult to reach in the network. The maximum degree k_{max} can also affect the consensus process, in other words, larger degrees make consensus easier.

		time for consensus	average degree	density	average clustering coef.	transitivity
time for consensus	Pearson Correlation	1	-.399**	-.957**	-.469**	-.575**
	Sig. (2-tailed)		.001	.000	.000	.000
	N	65	65	65	65	65
average degree	Pearson Correlation	-.399**	1	.537**	.608**	.562**
	Sig. (2-tailed)	.001		.000	.000	.000
	N	65	65	65	65	65
density	Pearson Correlation	-.957**	.537**	1	.539**	.661**
	Sig. (2-tailed)	.000	.000		.000	.000
	N	65	65	65	65	65
average clustering coef.	Pearson Correlation	-.469**	.608**	.539**	1	.835**
	Sig. (2-tailed)	.000	.000	.000		.000
	N	65	65	65	65	65
transitivity	Pearson Correlation	-.575**	.562**	.661**	.835**	1
	Sig. (2-tailed)	.000	.000	.000	.000	
	N	65	65	65	65	65

**. Correlation is significant at the 0.01 level (2-tailed).

Table 4.1.1: Correlations of time for consensus for all networks studied and their correspondent average degree, density, average clustering coefficient, and transitivity.

		time for consensus	diameter	algebraic connectivity	average pathlength	average distance-sum
time for consensus	Pearson Correlation	1	.810**	-.792**	.784**	.964**
	Sig. (2-tailed)		.000	.000	.000	.000
	N	65	65	65	65	65
diameter	Pearson Correlation	.810**	1	-.939**	.977**	.794**
	Sig. (2-tailed)	.000		.000	.000	.000
	N	65	65	65	65	65
algebraic connectivity	Pearson Correlation	-.792**	-.939**	1	-.935**	-.753**
	Sig. (2-tailed)	.000	.000		.000	.000
	N	65	65	65	65	65
average pathlength	Pearson Correlation	.784**	.977**	-.935**	1	.777**
	Sig. (2-tailed)	.000	.000	.000		.000
	N	65	65	65	65	65
average distance-sum	Pearson Correlation	.964**	.794**	-.753**	.777**	1
	Sig. (2-tailed)	.000	.000	.000	.000	
	N	65	65	65	65	65

**. Correlation is significant at the 0.01 level (2-tailed).

Table 4.1.2: Correlations of time for consensus for all networks studied and their correspondent diameter, algebraic connectivity, average path length, and average distance-sum.

From the inspection of the above correlations between the time for consensus and different structural characteristics, we can see that two main features appear to be important when trying to promote consensus: the way the nodes are connected and the total distance among them. For the case of the relation between the patterns of connections and the time necessary for reaching consensus, we know that those nodes with high degree may have a key role during the consensus process [14], thus we can relate the performance of a system when trying to reach agreement and its degree distribution. Degree distributions are well studied in the literature of network analysis (see references on page 19). We have a broad characterisation of networks according to the heterogeneity of these distributions, which may be, for example, power-law or Poisson-like distributions. There are works studying the relation between the ways nodes are connected, their degree distributions, and the time for reaching consensus [164–166], but to our knowledge, there are not works analysing the same for the case of distances. As we can see in the current chapter, distances among nodes also seem to play an important role for the consensus dynamics, so this characteristic deserves to be considered in a deeper way.

For the sake of illustrating the role that distances among nodes can play on the dynamics of consensus, let's consider the following example. Suppose that we have a network of n nodes with a structure of a path, just like the one shown in fig. 4.1.3, trying to reach agreement dictated by the model (see 2.3.2 on page 72)

$$x(t) = -Lx(\dot{t}),\ x(0) = x_0. \tag{4.1.6}$$

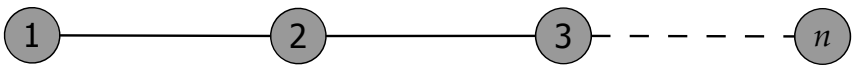


Figure 4.1.3: Illustration of a consensus process in a network with path-like structure

Let us assume that our network has node the 1 acting as a leader with an initial stationary state equal to 1, and the remaining $n - 1$ nodes acting as followers, with identical initial states equal to zero. While the process executes, in the first

time-step the information of node 1 has been shared with node 2 and this node has approached its state to the leader's one. In the second time-step, node 2 shares its new state with node 3, which in turn modifies its state to move towards the leader's state. This process continues for the next time-steps until a consensus is reached. Evidently, the time necessary for the system to reach consensus depends not just on the number of nodes, but on the distance among them, which for our case it increases as the number of nodes $n \to \infty$, thus playing an important role during the diffusion of the information.

4.2 Distance-sum distributions in complex networks.

Our previous results show that the total distance between any node and the rest of them is an important characteristic to be taken into account when studying complex networks and consensus processes. Thus, it is worth to devote some space to include a study of the distance-sum distributions on networks.

We start by defining the probability $p(s)$ of selecting uniformly at random a node with distance-sum s in a network [167]:

$$p(s) = \frac{n(s)}{n}, \tag{4.2.1}$$

where $n(s)$ is the number of nodes having distance-sum equal to s, and n is the size of the network. Then, the plot of $p(s)$ versus s represents the probability distribution function (PDF) of the distance-sum in a network. The cumulative distribution function (CDF) can be obtained by plotting the probability $P(s)$ of choosing at random a node with distance-sum larger or equal than s versus the distance-sum, where

$$P(s) = \sum_{s'=s}^{\infty} p\left(s'\right) \tag{4.2.2}$$

We studied the CDF instead of the PDF for the distance-sum of the nodes. The main reason is that the PDF is very noisy for both random and real-world networks, which makes difficult to find good fits for the distribution.

We start by studying several random networks with different degree distributions. The first class corresponds to the 'classical' random networks built by using the Erdös-Rényi (ER) model [168]. The second group corresponds to networks with power-law degree distributions, known as 'scale-free' (SF) networks [169], which were constructed by using the algorithm developed by Hagberg et al. [170]. In the ER graphs a group of nodes are connected randomly forming a graph with Poisson degree distribution. In the case of SF model, the resulting graph displays a power-law degree distribution of the type $p(k) \sim k^{-\gamma}$, where $p(k)$ is the probability of finding a node of degree k in the graph. We generated SF random networks with exponents $\gamma = 1.8,\ 2.5,\ 3.0$. The last ones are known as the Barabási-Albert (BA) networks [169, 171].

In figure 4.2.1 we illustrate the cumulative distance-sum distributions (CDSD) for the networks with the previously mentioned topologies and having 1000 nodes and average degree $\bar{k} = 8$. As we can see in this figure the shapes of the CDSDs for all the random networks studied here are very similar to each other. The best fits obtained for these cumulative distributions (displayed as a continuous line) correspond to cumulative normal distributions for a normal random variable with mean $\bar{s}$ and variance σ^2:

$$P\left(s, \bar{s}, \sigma^2\right) = \frac{1}{2}\left[1 + erf\left(\frac{s - \bar{s}}{\sigma\sqrt{2}}\right)\right], \tag{4.2.3}$$

where

$$erf\left(s\right) = \frac{2}{\sqrt{\pi}} \int_0^{\pi} e^{-t^2} dt. \tag{4.2.4}$$

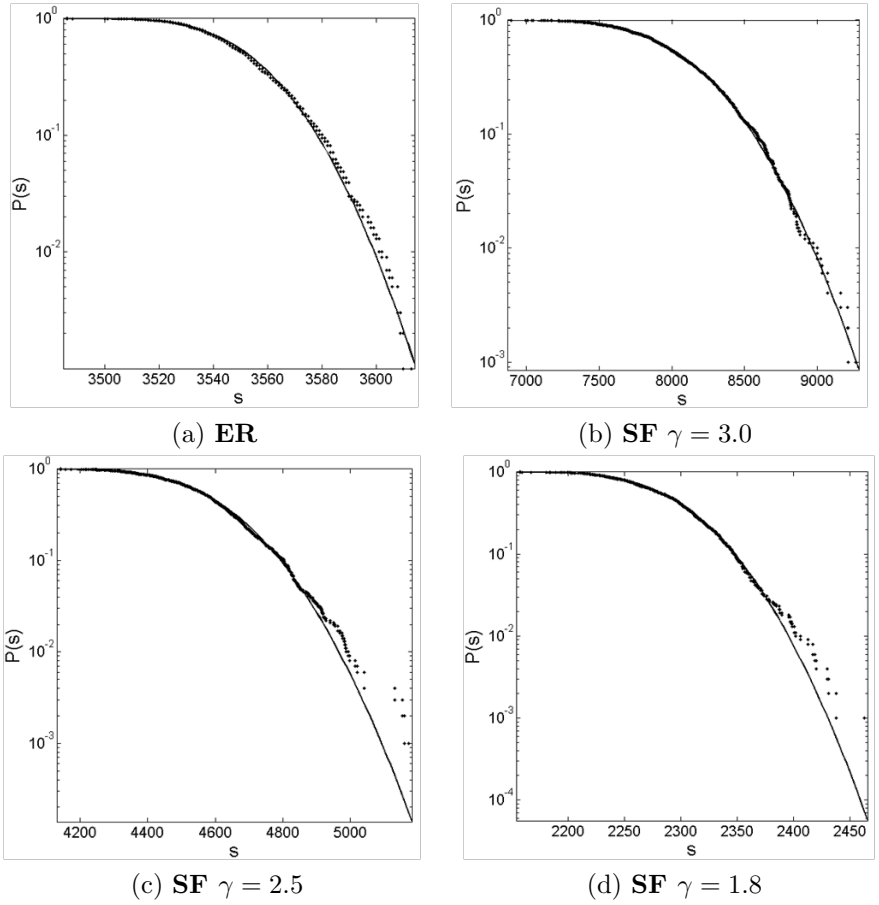
The fits in figure 4.2.1 were obtained by using the Distribution Fitting Tool© contained in Matlab© which uses the Maximum Likelihood Estimates (MLE) method to estimate the best parameters of a distribution for a given data. The

parameters for the best fits of all distributions in figure 4.2.1 are given in table 4.2.1.

The situation is more complex when we consider the cumulative distance-sum distributions of some real-world networks. For the sake of illustration we show in figure 4.2.2 the CDSD for the networks representing the food web of *Benguela*, a social network of injecting drug users in *Colorado Spring*, the food web of *Skipwith* pond and the protein-protein interaction of *Drosophila melanogaster*. The best fits found for such distributions are given in the same figure as solid lines. In no one case the best fit corresponds to the normal distribution but to Log-Logistic, Generalized extreme value, Weibull and Log-Normal distributions. The expressions for these cumulative distributions are given in table 4.2.2.

Network	μ	σ
ER $\left(\bar{k}=8\right)$	3552.34	20.19
SF($\gamma = 3.0$)	8053.17	393.96
SF($\gamma = 2.5$)	4580.65	165.75
SF($\gamma = 1.8$)	2288.79	45.89

Table 4.2.1: Fitting parameters for all CDSD of random networks and real networks represented in previous figure.



(a) **ER** (b) **SF** $\gamma = 3.0$

(c) **SF** $\gamma = 2.5$ (d) **SF** $\gamma = 1.8$

Figure 4.2.1: Cumulative distance-sum distributions (CDSD) for random networks with different topologies: (a) ER , (b) scale-free graphs with exponent 3.0, (c) exponent 2.5 and (d) exponent 1.8. The best fits for normal CDF are displayed as solid lines.

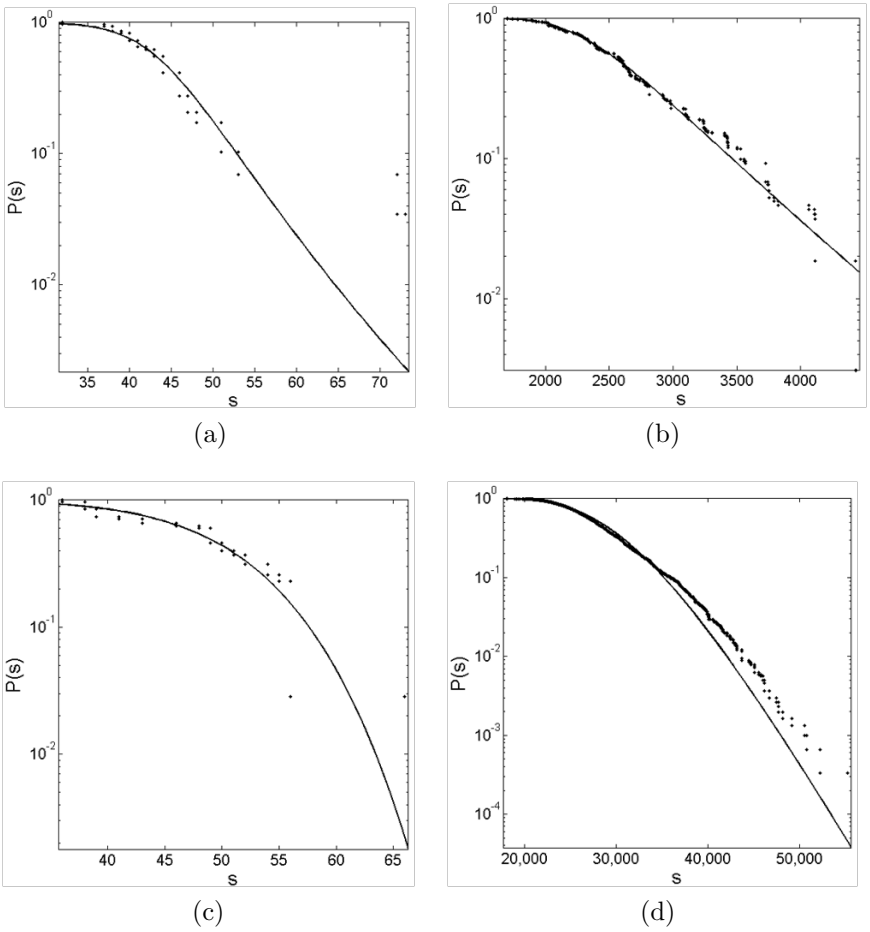


Figure 4.2.2: Cumulative distance-sum distributions (CDSD) for some real networks: (a) *Bengela* food web (b) social networks of injecting drug users at *Colorado Springs*, USA, (c) food web of *Skipwith* pond and (d) protein-protein interaction network of *Drosophila melanogaster*. The best fits for the normal CDF are displayed as solid lines, which correspond to Log-Logistic, Generalized extreme value, Weibull and Log-Normal, respectively.

Log-Logistic	Generalized extreme value
$P(s) = \frac{1}{1+\left(\frac{s}{\alpha}\right)^{-\beta}}$	$P(s) = e^{-t(x)},\ t(x) = \begin{cases} \left[1+\xi\left(\frac{s-\bar{s}}{\sigma}\right)\right]^{\frac{-1}{\xi}}, & \xi \in \mathbb{R} \\ e^{-\frac{(s-\bar{s})}{\sigma}} \end{cases}$
Weibull	Log-Normal
$P(s) = 1 - e^{-\left(\frac{s}{\lambda}\right)^k}$	$P(s) = \frac{1}{2} + \frac{1}{2} erf\left[\frac{\ln s - \bar{s}}{\sqrt{2\sigma^2}}\right]$

Table 4.2.2: Expressions for the cumulative distributions for (top-left) Bengela food web, (top-right) social network of injecting drug users at Colorado Springs, (bottom-left) food web of Skipwith pond, and (bottom-right) protein-protein interaction network of Drosophila melanogaster.

The analysis of the real-world networks provides a very good example of the difficulties that arise when statistical distributions are used as a way to quantify the distance-sum heterogeneity in networks. For instance, how can we compare distributions so mismatched as the ones found for only four real-world networks? Moreover, if we would like to set any relation with these distance-sum distributions and the times for consensus for the networks in fig. 4.2.2 ($t_{Bengela} = 257.70$, $t_{Colorado\,Springs} = 21946.34\, t_{Skipwith} = 85.18\, and\, t_{Drosophila} = 197708.37$), the problem is more challenging. This difficulty points out to the necessity of having an alternative way of quantifying this important characteristic. We tackle this situation in the next chapter by introducing an index for the distance-sum heterogeneity in networks.

Summary

This chapter provided an analysis of the correlation between some structural charateristics of different kinds of networks, and the time they need to reach consensus. We found that the average distance-sum presented the highest correlation value pointing to the fact that the distance is a key characteristic that deserves attention while studying consensus processes, thus we dedicated some space to study the distribution of distances in random graphs and complex networks. The results of the study on distance distributions showed the problems that arise when trying to fit well known distributions to those coming from real networks, and sets the grounds for the next chapter, where we propose a way to tackle such difficulties.

Chapter 5

DISTANCE-SUM HETEROGENEITY IN RANDOM GRAPHS AND COMPLEX NETWORKS

In chapter 3 we have seen that the effectiveness of those nodes acting as drivers (leaders) depends largely on the kind of structure the network presents, and even more, if this structure shows well defined communities the best strategy is to let those leaders to emerge from the local communities so that they can better promote consensus in the complete system, i.e., they are better distributed on the network. Chapter 4 showed that the total distance between any node and the rest of them is an important aspect for studying complex networks and consensus processes. The study of this characteristic is challenging since it is difficult to quantify the heterogeneity of distances by looking into their distributions, pointing to the necessity of having an alternative way for doing this. In this chapter we propose a new index to characterise the distance-sum heterogeneity of networks [167] which gives more insights about this important structural characteristic on random and real networks.

One important challenge for the study of complex networks is that many techniques developed for the analysis of small graphs are computationally intractable for

gigantic complex networks found in the real-world. On the other hand, some statistical approaches developed so far for the analysis of these huge networks are not applicable to small graphs. An example of the last situation is the analysis of degree heterogeneity in networks [172], which is frequently carried out by studying the distributions of node degrees [169, 173, 174]. In large networks it is possible to analyse the distribution of the probabilities $p(k)$ of finding a node with degree k as a function of the node degree, but if in this network there are only nodes with degree a and b we would have only two points for the distribution and this would be meaningless. However, in a small graph there is not enough data points as for having a good fit for these distributions. Other difficulties found in studying degree distributions include the selection of the best fit, and the way to compare the heterogeneity of networks with different types of distributions [172, 174].

Statistical distributions of other graph-theoretic parameters have also been studied for complex networks, such as the eigenvalue distributions [175] and the node-node distance distribution [176–178]. The distance-based analogous of the node degree distribution in a network is the distance-sum distribution. This kind of distribution has not previously been studied for networks. It consists of the distribution of the probabilities $p(s)$ of finding a node with distance-sum equal to s, where s is the sum of all distances from a given node as defined in 1.1.10 on page 24.

Distance-sum is an important characterisation of a node that can be found in many graph-theoretic invariants such as the Wiener [179] and Balaban [180] indices, average shortest path [34] and closeness centrality [181].

The Wiener index is defined as follows [179, 182]

$$W(G) = \sum_{i} \sum_{j>i} d(i,j) = \frac{1}{2} \sum_{i=1}^{n} s(i) . \tag{5.0.1}$$

The Balaban index is defined as [180]

$$J(G) = \frac{m}{C+1} \sum_{(i,j)\in E} (s_i s_j)^{-\frac{1}{2}} , \tag{5.0.2}$$

where $C = m - n + 1$ is the cyclomatic number.

The average path length is defined as [5]

$$\bar{l} = \frac{\sum_{i=1}^{n} s\left(i\right)}{n\left(n-1\right)} = \frac{2W(G)}{n\left(n-1\right)}. \tag{5.0.3}$$

The so-called 'small-world' effect is present in a given network when $\bar{l}$ is small compared to the size of the network, i.e., $\bar{l} \sim \ln n$ [34]. The small-world effect impacts directly on the properties of networked systems and dynamical processes in networks, particularly those related with communications and synchronization [183].

Another graph-theoretic measure related to the distance-sum of a given node is the closeness centrality (3.2.3 on page 95) $CC\left(i\right) = \frac{n-1}{\sum_{j \in V(G)} d(i,j)} = \frac{n-1}{s(i)}$,which characterizes how close a node is from the rest of nodes in a network [181, 184].

In a similar way to the analysis of any kind of statistical distributions for the nodes of a graph, distance-sum distributions are difficult to find for small graphs where the number of data points is very scarce as well as the other difficulties mentioned before. In those cases where the distributions can be found the previously stated difficulties for analysing the heterogeneity of networks also apply to the analysis of distance-sums. Consequently, we propose here the derivation of an index quantifying the distance-sum heterogeneity of a graph/network in such a way that it can be applicable for a graph of any size.

5.1 Distance-sum heterogeneity in graphs and complex networks

We start by establishing the grounds for our analysis. We consider a simple, undirected and unweighted graph $G = (V, E)$ with $n = |V|$ vertices and $m = |E|$ edges. This graph has an associated square and symmetric adjacency matrix $\mathbf{A}$ with entries 0 and 1 as defined in 1.2.1 on page 26. The graph has an associated Laplacian matrix $\mathbf{L} = \boldsymbol{K} - \boldsymbol{A}$, as defined in 1.2.14 on page 30. We remember that the Laplacian is positive semidefinite with eigenvalues $0 = \mu_1 < \mu_2 \leq \cdots \leq \mu_n$ for a connected graph.

Recall that the degree of the node i is given by $k_i = \sum_{j=1}^{n} A_{ij}$ and that the density of a graph is defined as $\varrho = \frac{2m}{n(n-1)}$. We also know that among all the paths between two nodes v_1 and v_{l+1}, the ones having the minimum length are called *shortest-paths*, and that the length of a shortest path between v_i and v_j is called the shortest-path distance (or simply the distance $d_{i,j}$) between nodes v_i and v_j .

As defined on page 24, the distance matrix $\mathbf{D}$ of the graph G is a square, symmetric $n \times n$ matrix whose i, j entry is given by $d_{ij} = d(i, j)$. The *distance-sum* $s(i)$ of a node i is defined as , and a vector of distance-sums can be obtained as $\mathbf{s} = \mathbf{1}^T\mathbf{D}$, being $\mathbf{1}$ a column vector of ones.

5.1.1 Distance-sum heterogeneity index

In order to tackle the problem previously set on page 132, we propose a new approach to quantify the distance-sum heterogeneity of networks which is encoded in what we have called the *distance-sum heterogeneity index*. This index shows some important differences in the distance-sum heterogeneity of random and real-world networks.

To introduce the distance-sum heterogeneity index we start by considering a hypothetical process in which the nodes of a given network reach a consensus about their distance-sums. That is, let $G = (V, E)$ be a simple, undirected and unweighted graph with distance-sum of the nodes given by the vector $\mathbf{s}$. Let $f(s_i)$ be a function of the distance-sum of node i. In the hypothetical consensus process every pair of connected nodes tries to 'equalize' their functions $f = f(s_i)$ of distance-sums by a consensus process. The final consensus state is reached if, for all $f_i(0)$ and all $i, j = 1, \ldots, n$, $|f_i(t) - f_j(t)| \to 0$ as $t \to \infty$ [7]. We know that the consensus model has the form $\frac{d\mathbf{f}}{dt} = -\mathbf{L}\mathbf{f}$, $\mathbf{f}(0) = \mathbf{f}_0$, where $\mathbf{L}$ is the Laplacian matrix of the network. In order to control the evolution of the consensus process in the network a disagreement function $\varphi(f)$ is defined as [7]

$$\varphi(f) = \frac{1}{2}\mathbf{f}^T\mathbf{L}\mathbf{f} = \frac{1}{2}\langle f | \mathbf{L} | f \rangle \tag{5.1.1}$$

such as that the consensus model can be written as the gradient-descent algorithm [7]

$$\frac{d\mathbf{f}}{dt} = -\nabla\varphi\left(f\right), \mathbf{f}\left(0\right) = \mathbf{f}_0 \tag{5.1.2}$$

Now, returning to the quadratic form (5.1.1), we remark that it can be written as

$$\varphi\left(f\right) = \frac{1}{2}\sum_{(i,j)\in E}\left(f_i - f_j\right)^2 \tag{5.1.3}$$

indicating that $\varphi(f)$ measures the 'heterogeneity' in the distance-sum function f in every time-step of the consensus process.

Here we are not concerned with the time evolution of the 'heterogeneity' function in the consensus process, but mainly on how much heterogeneity a given graph has. That is, we are interested in finding $\varphi(f)$ only for time zero of the consensus process. For the sake of convenience we select our function f to be a power of the distance-sum, i.e., $f = f\left(s_i\right) = s_i^{\alpha}$. This is a general form that can embrace such indices like the Wiener index and closeness centrality ($\alpha = 1$), as well as the Balaban index ($\alpha = -\frac{1}{2}$) one. In closing, the distance-sum heterogeneity of a graph is given by the following formula:

$$\varphi\left(G\right) = \frac{1}{2}\sum_{(i,j)\in E}\left(s_i^{\alpha} - s_j^{\alpha}\right)^2 = \frac{1}{2}\left(s^{\alpha}\right)^T \mathbf{L}s^{\alpha} \tag{5.1.4}$$

5.1.1.1 Properties of the distance-sum heterogeneity index

For the following analysis of the distance-sum heterogeneity index (5.1.4) we will consider only the case $\alpha = -\frac{1}{2}$, which relates the our proposed index with the Balaban index for a given graph.

Let $\varphi\left(G\right)$ be the heterogeneity index of a simple, undirected, unweighted graph G and let $\alpha = -\frac{1}{2}$. Then, it can be easily shown that

$$\varphi\left(G\right) = \sum_{i=1}^{n}\frac{k_i}{s_i} - 2\sum_{(i,j)\in E}\left(s_i s_j\right)^{-\frac{1}{2}}, \tag{5.1.5}$$

where k_i is the degree of the node i. Note that the term in the second part of the right-hand side of (5.1.5) corresponds to the Balaban index except for the factor $\frac{m}{(C+1)}$.

The term $\frac{k_i}{s_i}$ in the expression (5.1.5) has the following interpretation. Let us consider a walker living at node i who can visit every node j of the connected graph by using the shortest paths from i to j. Let us consider a discrete-time process in which the time needed by the walker for going from one node to a nearest neighbour is $t = 1$. Here we consider independent visits to the nodes of the graph. That is, if a walker at node i visits the node j at distance d_{ij} it is assumed that the walker returns to i before he visits another node v. Thus, the total time needed by a walker living at node i for independently visiting every node of the network is $t_T(i) = 2s_i$. On the other hand, the time needed for independently visiting every nearest neighbour of node i is given by $t_{nn}(i) = 2k_i$. Consequently, the fraction of the total time needed by the walker to independently visiting all his nearest neighbours is given by:

$$r_t(i) = \frac{t_{nn}(i)}{t_T(i)} = \frac{k_i}{s_i}, \tag{5.1.6}$$

which defines a new centrality index for the nodes of a network.

If we consider n walkers living at the n nodes of a network, the average fraction of time needed by them to independently visiting their nearest neighbours is $r_t = \frac{1}{n}\sum_{i=1}^{n} r_t(i)$. Using these expressions we can rewrite the Balaban $J(G)$ index [185] in terms of the average fraction of time r_t and the distance-sum heterogeneity index as

$$J(G) = \gamma\left[nr_t - \varphi(G)\right], \tag{5.1.7}$$

where $\gamma = \frac{2m}{(C+1)}$.

It is evident from (5.1.4) that the lower bound for the distance-sum heterogeneity index is zero, which is reached when the graph has the value of s_i for every node. In order to search for the maximum of this index a computation-based search among all connected graphs with 4 to 24 nodes was carried out using the

AutoGraphiX system for finding extremal graphs [186] which uses the variable neighbourhood search metaheuristic to complete the task as a problem of combinatorial optimisation. The variable neighbourhood search differentiates itself from other methods for solving combinatorial optimisation problems, i.e. simulated annealing, tabu search and genetic algorithms, in the sense that the last methods perform a sequence of local changes in an initial solution which improve the value of the objective function each time until a local optimum is found [187], while the first one does not follow a trajectory, but explores increasingly distant neighbours of the current incumbent solution, jumping from this to a new one if and only if an improvement is reached, thus avoiding being trapped in local optima [188].

For graphs with $n = 4, 5$ the maximum of the distance-sum heterogeneity index is reached for the star graph. For graphs with $n = 6, 7, 8$ the maximum is reached for the graphs having the structures illustrated in figure 5.1.2. These graphs are easily constructed from a star graph $S_{1,n-1}$ by making a duplicate copy of the node having degree $n - 1$. These type of graphs have been called 'agave' in allusion to the plant from which Tequila is produced [167]. For graphs with n =9 to 24, the maximum is given by a structure called *complete split graphs* [189], like the ones shown in fig. 5.1.3. The parameters for these kind of graphs that maximise the distance-sum heterogeneity index $\varphi(G)$ are given in table 5.1.1.

n	4	5	6	7	8	9	10	11	12	13	14
α	3	4	4	5	6	6	7	8	9	9	10
n	15	16	17	18	19	20	21	22	23	24	
α	11	11	12	13	14	14	15	16	16	17	

Table 5.1.1: Parameters n and α for which the complete split graph $SK_{n,\alpha}$ maximises the distance-sum heterogeneity index for $3 \leq n \leq 24$. These values were obtained with the AutoGraphiX system.

A complete split graph $SK_{n,\alpha}$ is the graph obtained from an empty graph on α vertices, $\bar{K}_\alpha$, and a clique on $n - \alpha$ vertices, $K_{n-\alpha}$, by adding all possible edges between the independent set and the clique (see for example fig. 5.1.3). We point out to the fact that in the case where $\alpha = n-2$, the complete split graph $SK_{n,n-2}$ gives the agave graph. For this particular case, an agave graph with n vertices, the distance-sum heterogeneity index is given by

$$\varphi(SK_{n,n-2}) = (2n-4)\left(\frac{1}{\sqrt{n-1}} - \frac{1}{\sqrt{2n-4}}\right)^2 = \frac{3n-5}{n-1} - 2\left(2 - \frac{2}{n-1}\right)^{\frac{1}{2}}. \tag{5.1.8}$$

For the case of the distance-sum heterogeneity for the complete split graph, let us use the illustration on fig. 5.1.1. We know that the distance-sum heterogeneity index for any graph $G(V, E)$ is given by

$$\varphi(G) = \frac{1}{2}\sum_{(i,j)\in E}\left(\frac{1}{\sqrt{s_i}} - \frac{1}{\sqrt{s_j}}\right)^2. \tag{5.1.9}$$

Lemma 5.1. *The distance-sum heterogeneity index of $SK_{n,\alpha}$ is $\varphi(SK_{n,\alpha}) = \alpha(n-\alpha)\left(\frac{1}{\sqrt{n-1}} - \frac{1}{\sqrt{2(n-1)-\alpha}}\right)^2$*

Proof. For a complete split graph $SK_{n,\alpha}$ like the one in fig. 5.1.1, we can identify two types of nodes:

1. $n - \alpha$ nodes which are part of the clique (dark grey nodes).

2. α nodes which are not part of the clique (light grey nodes).

Thus, we have that

$$s_i = \alpha + 2(n-\alpha-1) = \alpha + 2n - 2\alpha - 2 = 2n - \alpha - 2 = 2(n-1) - \alpha, \tag{5.1.10}$$

$$s_j = (n-\alpha-1) + \alpha = n - 1, \tag{5.1.11}$$

and taking the expressions 5.1.13 and 5.1.14 into 5.1.9 leads to

$$\varphi(SK_{n,\alpha}) = \alpha(n-\alpha)\left(\frac{1}{\sqrt{n-1}} - \frac{1}{\sqrt{2(n-1)-\alpha}}\right)^2. \tag{5.1.12}$$

□

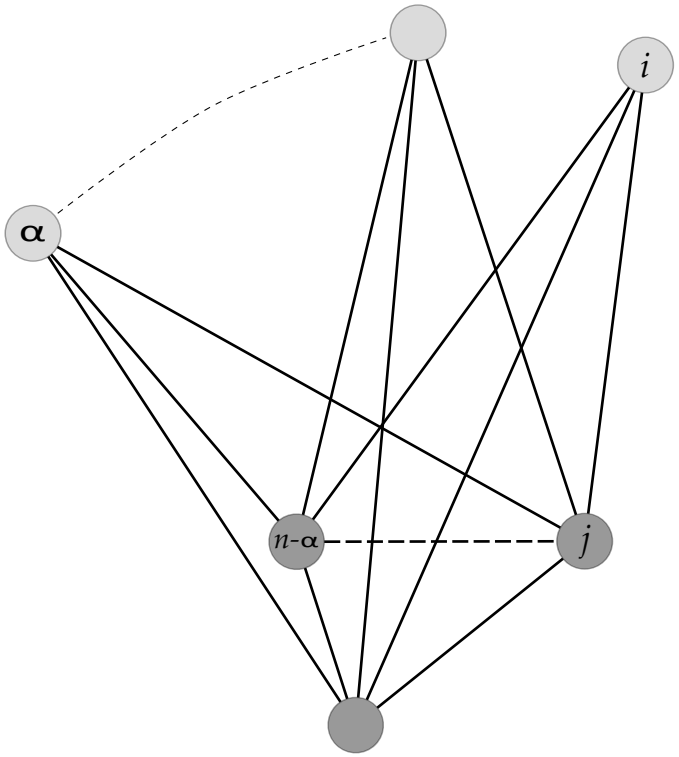


Figure 5.1.1: Illustration of a complete split graph. The dark grey nodes form a clique of size $n - \alpha$. The α light grey nodes are connected to the clique but are not connected among them.

The structure of a complete split graph (see fig. 5.1.1) has adjacency matrix given by the following general structure

$$\mathbf{A}(SK_{n,\alpha}) = \begin{bmatrix} \mathrm{K}_{n-\alpha} & \mathrm{J}_{n-\alpha,\alpha} \\ \mathrm{J}_{\alpha,n-\alpha} & 0_{\alpha,\alpha} \end{bmatrix}, \tag{5.1.13}$$

and their correspondent Laplacian matrix $\boldsymbol{L}(SK_{n,\alpha})$ has a spectrum

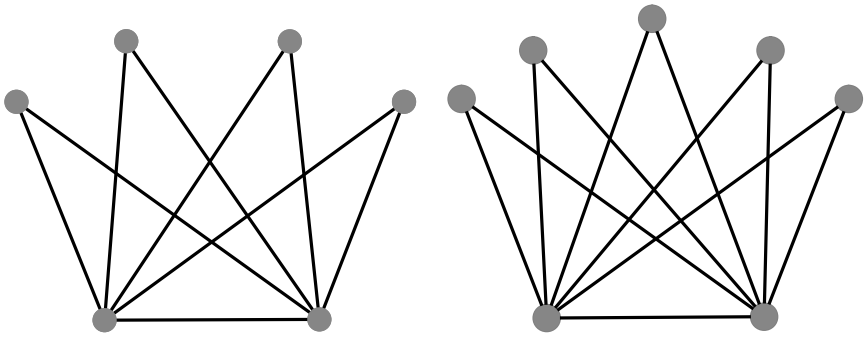
$$SK_{n,\alpha} : \ Sp(\mathbf{L}) = \{\mu_1(\mathbf{L}),\, (\alpha-1)(\mu_2(\mathbf{L})),\, (n-\alpha)(\mu_3(\mathbf{L}))\}\,, \tag{5.1.14}$$

where $\mu_1 = 0$.

Let α^* be the value for which $\varphi(SK_{n,\alpha})$ is maximum for $\alpha \in \{1,\, 2, \ldots,\, n-1\}$. An inspection of the values in table 5.1.1 gives us that $\alpha^* = \left\lceil \sqrt{\frac{n(n-1)}{2}} \right\rceil$. We conjecture

that the complete split graph maximizes the distance-sum heterogeneity index for graphs having $n \geq 4$ nodes.

Conjecture 5.2. *Among the graphs having $n \geq 4$, the complete split graph SK_{n,α^*} has the maximum distance-sum heterogeneity index.*



(a) (b)

Figure 5.1.2: Illustration of the agave graphs with (a) 6 nodes and (b) 7 nodes.

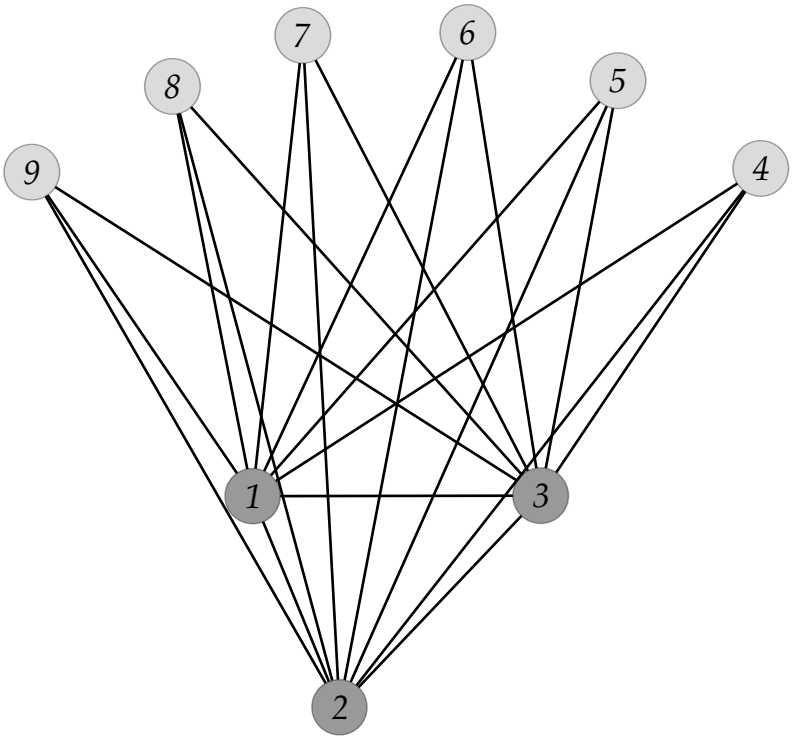


Figure 5.1.3: Illustration of the complete split graphs with 9 nodes.

The distance-sum heterogeneity index can be expressed in terms of the 'optimal'

values of the index given by the SK_{n,α^*} graphs. The following formula expresses the distance-sum heterogeneity as a percentage of the conjectured maximum possible value of distance-sum heterogeneity:

$$\varphi_{rel} = \frac{100 \cdot \varphi(G)}{\varphi_{opt}(G)}. \tag{5.1.15}$$

5.1.2 Spectral representation of the distance-sum heterogeneity index

We study the spectral representation of the distance-sum heterogeneity index. We start by considering the $\mathbf{u}_j$ orthonormal eigenvector of the Laplacian matrix associated with the μ_j eigenvalue. The cosine of the angle formed between this eigenvector and the vector of distance-sum $\mathbf{s}^{-\frac{1}{2}}$ for a given network is expressed as

$$\cos\theta_j = \frac{\mathbf{s}^{-\frac{1}{2}} \cdot \mathbf{u}_j}{\left\|\mathbf{s}^{-\frac{1}{2}}\right\|}, \tag{5.1.16}$$

where $\left\|\mathbf{s}^{-\frac{1}{2}}\right\|$ is the Euclidean norm that can be written as $\left\|\mathbf{s}^{-\frac{1}{2}}\right\| = \sqrt{\sum_i s_i^{-1}}$. Let ${}^0J_{-1} = \sum_i^n s_i^{-1}$. Then, using the Euler theorem (see page 457 of ref. [190]) we can represent the distance-sum heterogeneity index in terms of the eigenvalues of the Laplacian and the cosines θ_j as follows

$$\varphi(G) = \frac{1}{{}^0J_{-1}} \sum_{j=2}^{n} \mu_j \cos^2\theta_j. \tag{5.1.17}$$

The term $\cos^2\theta_j$ represents the similarity between the normalised distance-sum vector and the corresponding eigenvector (or vice versa). For instance, $\cos^2\theta_j = 0$ means that the vector $\mathbf{s}^{-\frac{1}{2}}$ is perpendicular to the Laplacian eigenvector $\mathbf{u}_j$, and no "duplicated" information is contained in both vectors, which means that they are dissimilar.

Now we can consider a graphical representation of the distance-sum heterogeneity of a network if we take a coordinate system with origin at $\mu_1 = 0$. We can represent the other eigenvalues of the Laplacian for a given network as points in this system in the following way: we consider that the eigenvector $\mu_{j>1}$ is represented by a point whose distance from the origin of coordinates O is given by $B = \sqrt{\mu_{j>1}}$. The segment OB forms an angle θ_j with the y axis of coordinates, which determines the full position of the point in the coordinate system. It can be seen that the projection of $\sqrt{\mu_{j>1}}$ on the x axis is given by $x_j = \sqrt{\mu_{j>1}} \cos\theta_j$, and the projection of $\sqrt{\mu_{j>1}}$ on the y axis is given by $y_j = \sqrt{\mu_{j>1}} \sin\theta_j$. This means that the distance-sum heterogeneity index $\varphi(G)$ can be written as

$$\varphi(G) = \left({}^0J_{-1}\right)^{-1} \sum_{j=1}^{n} x_j^2. \tag{5.1.18}$$

We can use this kind of plot to represent the distance-sum heterogeneity of a network in a graphical form by plotting x_j vs. y_j for all values of j. Thus the distance-sum heterogeneity index is given by the sum of the squares of the projections of all these points on the abscissa. Obviously, all projections on y-axis are positive but those on x-axis can have positive and negative signs. We call these plots distance-sum heterogeneity plots or simply *S-plots*.

5.1.3 Distance-sum heterogeneity index for random and real-world networks

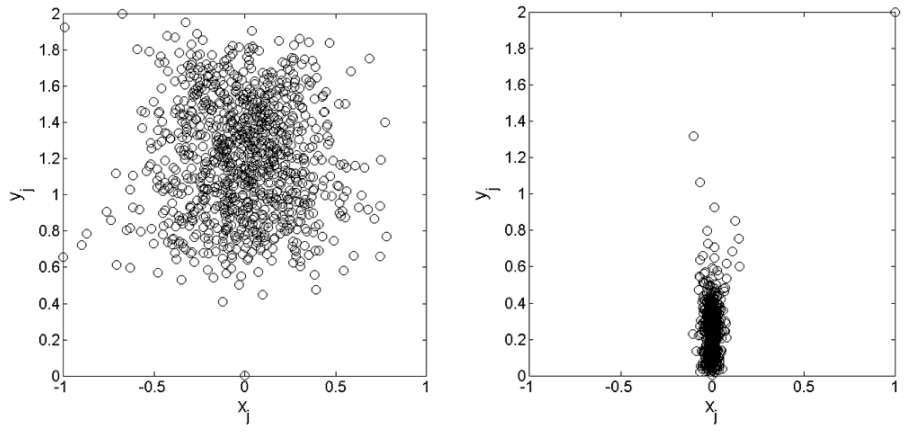
5.1.3.1 Distance-sum heterogeneity in random networks

As we previously showed, the cumulative distance-sum distributions for random graphs do not display any significant difference in the distance-sum heterogeneity of networks with quite different topologies. We calculated the distance-sum heterogeneity index for the same random graphs displayed in figure 4.2.1 as well as their values relative to the conjectured maximum heterogeneities and the results are given in table 5.1.2.

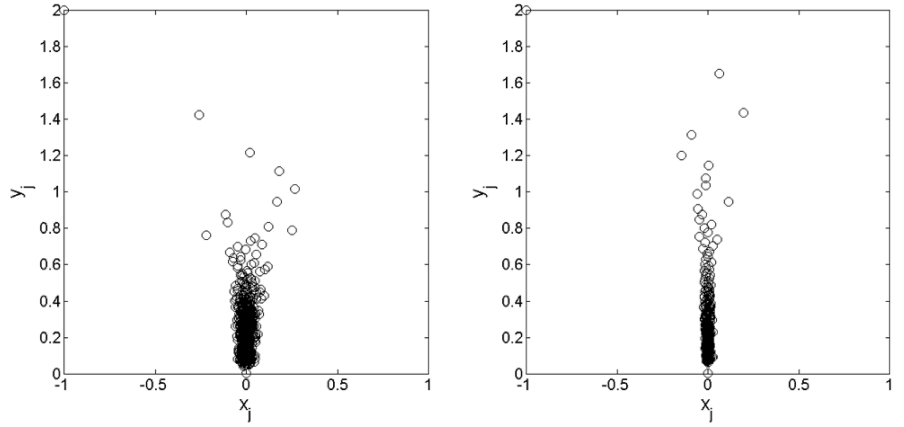
Random Network	$\varphi(G)$	$\varphi_{rel}(G)$
SF $\gamma = 1.8$	0.05487	4.8102
SF $\gamma = 2.5$	0.00684	0.0599
SF $\gamma = 3.0$	0.00295	0.0258
ER	0.00103	0.0090

Table 5.1.2: Distance-sum heterogeneity index and their relative values for different random graphs.

As we can see in table 5.1.2, the degree distribution induces distance-sum heterogeneity in the random networks. The ER network displays the lowest distance-sum heterogeneity with a value of $\varphi_{rel}(G)$ close to zero. This indicates that most of the nodes in an ER network have approximately the same distance-sum displaying a remarkable regularity. As soon as the degree distribution becomes more skewed there are some nodes that concentrate much more links than the rest, i.e., the hubs of the networks. As a consequence, the hubs have a larger number of small shortest paths than the poorly-connected nodes. This unbalance makes that the distance-sum heterogeneity increases in these networks.



(a) **ER** (b) **SF** $\gamma = 3.0$



(c) **SF** $\gamma = 2.5$ (d) **SF** $\gamma = 1.8$

Figure 5.1.4: S-plots for different random networks: (a) ER, (b) scale-free with exponent 3.0, (c) scale-free with exponent 2.5, and (d) scale-free with exponent 1.8.

Graphically, these heterogeneities can be better observed by using the S-plots for these networks. In figure 5.1.4 we illustrate the S-plots for the random networks studied and it can be seen that the ER network has a very homogeneous S-plot, which covers practically all the values of x_j in the interval $[-1, 1]$. The networks with SF topologies display very narrow S-plots in which most of the x_j values are concentrated around the zero value. A further characterization of these plots would add more value to the analysis of distance-sum heterogeneity in networks. However, we will not consider such kinds of quantitative characterizations in this thesis.

5.1.3.2 Distance-sum heterogeneity in real-world networks

Now we study the distance-sum heterogeneity of 64 real-world networks representing biological (B), ecological (E), informational (I), social (S) and technological (T) systems. The description of all these networks can be found in appendix B on page 164.

Among these networks, we have biological networks that include the neural network of *C. elegans*; the transcription networks of yeast, *E. coli* and urchins; the PPI networks of *D. melanogaster*, *H. pylori*, *A. fulgidus*, *B. subtilus*, *E. coli*, *malaria parasite*, *Kaposi sarcoma herpes* virus, *human* and *yeast*. Ecological networks include food webs *like Benguela*, *Coachella Valley*, *Reef Small*, *Shelf*, *Skipwith* pond, *St. Marks* seagrass, *Stony* stream, *Bridge Brook*, *Canton Creek*, *Chesapeake* Bay, *El Verde* rainforest, *Scotch Broom*, Grassland *Little Rock*, *St. Martin* and *Ythan* estuary with and without parasites. Informational networks represent systems such as a network of the *Roget thesaurus*; a citation network consisting of papers published in the *Proceedings of Graph Drawing* in the period 1994–2000; a semantic network of the Online Dictionary of Library and Information Science (*ODLIS*); a citation network in the field of "small-world". Among those social networks considered we have the social networks of corporate elite in USA, inmates in prison, the friendship network between physicians (*Galesburg*), the friendship ties among the employees in a small hi-tech computer firm which sells, installs, and maintains computer systems (*high-tech*), and a sawmill communication network; the social networks of injecting drug users, a social network among college students in a course about leadership and the *Zachary* karate club; persons with HIV infection during its early epidemic phase in *Colorado Springs*, a scientific collaboration network in the field of computational geometry, and a sexual network consisting of heterosexual relationships. Finally, among the technological networks we included three electronic sequential logic circuits parsed from the ISCAS89 benchmark set; the software network of *Abi, Digital, MySQL, VTK* and *XMMS*; the USA airport transportation network of 1997; two versions of Internet at autonomous system of 1997 and 1998.

According to our calculations, real-world networks do not display very large distance-sum heterogeneity indices. The average value of the relative distance-sum heterogeneity is about 2.5%. However, there are significant variations of this index for

individual networks. For instance, the food webs of Skipwith and Bridge Brooks have relative distance-sum heterogeneities of 29% and 11.5%, respectively, and the citation network of 'small-world' has a value of near 1.5%. On the other side of the coin there are 3 networks with relative distance-sum heterogeneities smaller than 0.009%, which are accordingly very close to those observed for random networks with Poisson degree distributions. In figure 5.1.5 we illustrate the values of the relative distance-sum heterogeneity indices for all the real-world networks studied.

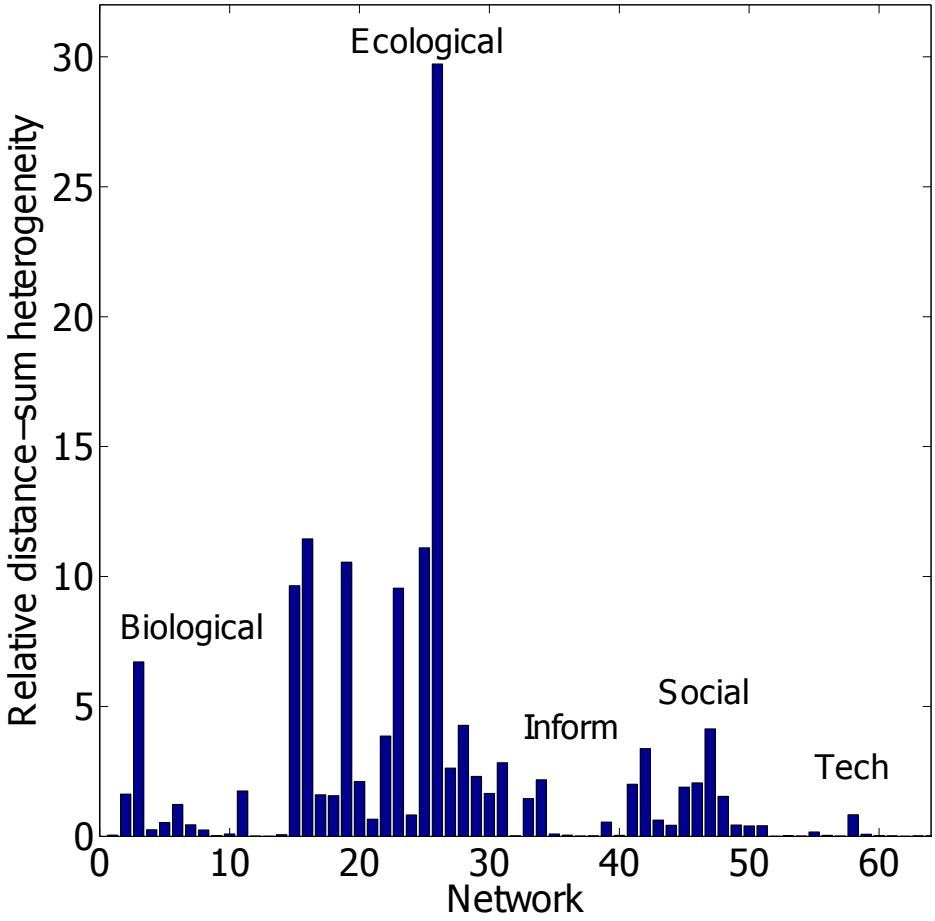


Figure 5.1.5: Values of relative distance-sum heterogeneity indices for the 64 real-world networks studied.

When the average relative distance-sum heterogeneity is considered for all networks in the different functional classes, i.e., B, E, I, S and T, we find some interesting observations. First, the largest distance-sum heterogeneity is observed for the

ecological networks, which display an average of about 6% of the conjectured maximum value for this index. The removal of the two networks with the largest distance-sum heterogeneity does not change very much this situation. For instance, after removing the food webs of Skipwith and Bridge Brooks the remaining food webs have an average of 4% of relative distance-sum heterogeneity, which is still four times of those observed for the networks in the other groups. Technological networks have average relative distance-sum heterogeneity of about 0.10% and the other three groups are very close to each other with percentages between 0.75% (I) and 1.12% (S) (see figure 5.1.6).

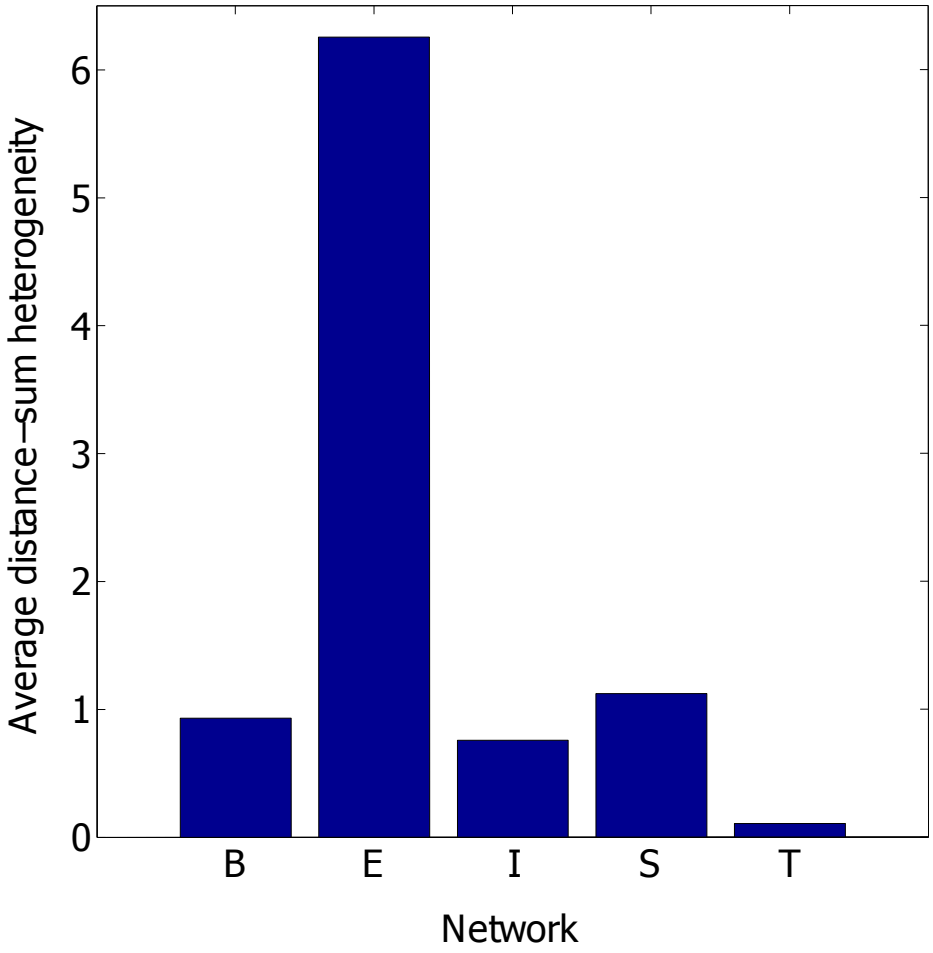
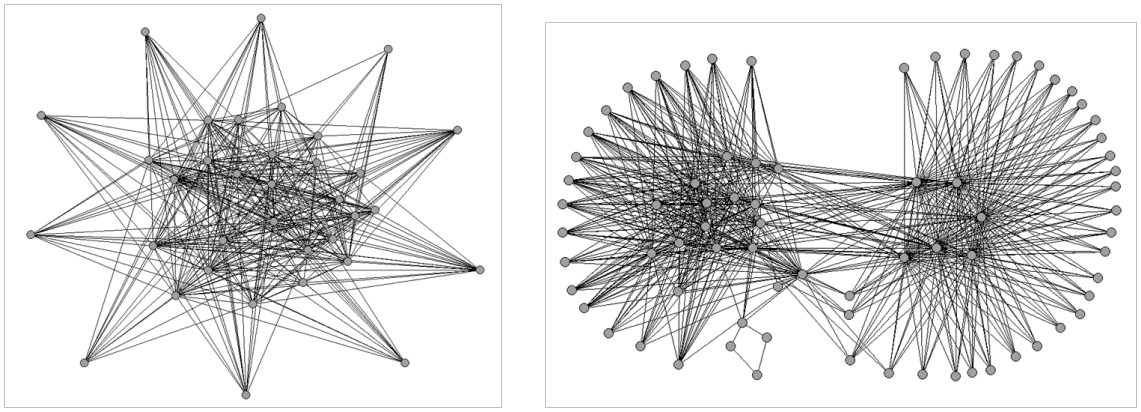


Figure 5.1.6: Average relative distance-sum heterogeneity for all networks grouped into different functional classes: Biological (B), Ecological (E), Informational (I), Social (S) and Technological (T).

The reason why food webs display more relative distance-sum heterogeneity is

not clear. These are networks with higher densities than the rest of the networks studied here. For instance, the densities of the food webs analysed are about 6 times larger than the average of the rest of the networks. As a consequence, there is a linear correlation between the density and the relative distance-sum heterogeneity of the networks studied. It displays a Pearson correlation coefficient of 0.92 indicating that the denser networks are also the most distance-sum heterogeneous. However, it is easy to be fooled by this kind of correlations as we can build networks with high density and very poor distance-sum heterogeneity (think for instance about the complete graph). In fact, if we remove all food webs from the previous correlation, the correlation coefficients drops to 0.89, indicating that there is no such kind of strong dependence and that the previous observation appears to be biased by the presence of food webs. Thus, it is plausible that there is some kind of functional cause for the appearance of distance-sum heterogeneity in food webs. A review of two examples of networks with some of the largest relative distance-sum heterogeneities gives some important hints. Lets take for example, the networks *Skipwith* and *Bridge Brooks*. It is evident from figure 5.1.7 that these food webs resemble very much the type of graphs we have conjectured to display the largest values of distance-sum heterogeneity. This type of structure can appear naturally in the evolution of food webs, where there could be a central core of species with trophic relations among them, surrounded by one or more layers of species that have trophic relations with the central core but not among them. This could be the case, for example, of parasites that have trophic interactions with other species but not among them. In closing, we have found that the type of topological structure that maximizes the distance-sum heterogeneity of any graph can appear naturally in ecological food webs, where they can explain some of the structural and dynamical properties of such ecological systems.



(a) (b)

Figure 5.1.7: Illustrations of two food webs with some of the largest relative distance-sum heterogeneities: (a) *Skipwith* and (b) *Bridge Brooks*. Nodes represent species and links represent trophic interactions (who-eat-who) in the ecosystem.

We also explored the relationship between the relative distance-sum heterogeneity index and the average path length. We have found that the relative distance-sum heterogeneity index decays as a power-law with the average distance. Accordingly, $\varphi_{rel}(G) = 0.7434 \cdot \bar{l}^{-2.126}$ with correlation coefficient equal to 0.91 (see fig. 5.1.8). This relationship indicates that the networks with large relative distance-sum heterogeneity have small average path length. From an inspection to the form of the complete split graph, we can infer that the star graph has the largest average shortest path distance, and that the star is the initial stage for the generation of graphs with maximum distance-sum heterogeneity, and that in every further step we are adding links in a way that decreases the average distance among nodes. For example, the average path length in the star graph with n nodes is given by

$$\bar{l}(S_n) = 2 - \frac{2}{n}. \tag{5.1.19}$$

It is straightforward to realize that $\bar{l}(S_n) \to 2$ as $n \to \infty$, and as the density of the graphs increases the average path length drops quickly. Consequently, the graphs which have high relative distance-sum heterogeneity necessarily have small average path length.

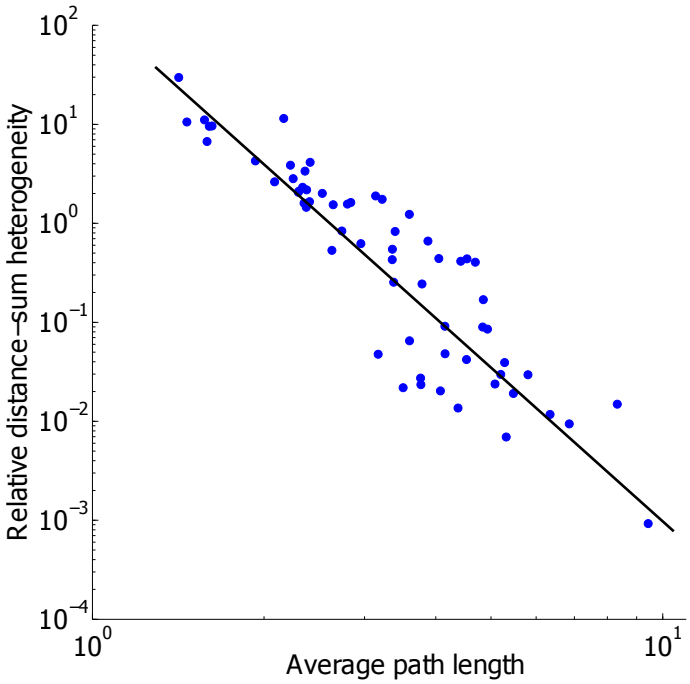


Figure 5.1.8: Relation between the relative distance-sum heterogeneity index and average shortest path distance for the 64 real-world networks studied. The plot is in log-log scale to illustrate the power-law relationships existing between both parameters.

5.2 Distance-sum heterogeneity and consensus in complex networks

As stated in chapter 4, the total distance among nodes in a network plays a key role when trying to reach consensus, but since we lacked of a metric that could give some insights about this characteristic, we derived an index that can be used now for testing this hypothesis. First we plot the times for consensus for our 64 networks versus their relative distance-sum heterogeneity obtained with the expression 5.1.15. For analysis purposes we normalised the times for consensus by the number of nodes n so we can compare two normalised properties. As we can see in fig. 5.2.1, these variables show a decay close to the form $t_{rel} = 1.0976\varphi_{rel}^{-0.5686}$, with correlation coefficient of 0.7. This relation points to the idea that consensus is better achieved for those networks with larger distance-sum heterogeneity. This result might seem counter intuitive, but if we recall from the previous section, large

relative distance-sum does not imply necessarily that the the distance among nodes is large, it implies that the difference in the total distance among nodes is large, which could be the case that there may be nodes strongly connected to other nodes, but those nodes may not be connected among them, this is the case of the structure of a complete split graph, which was conjectured it maximises the distance-sum heterogeneity index $\varphi(G)$.

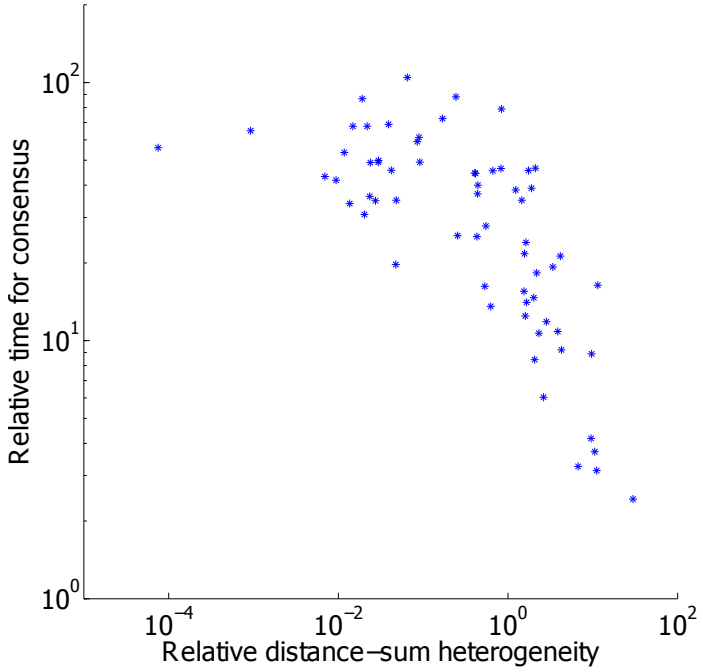


Figure 5.2.1: Log-log plot of relative time for consensus versus the relative distance-sum heterogeneity for the 64 networks studied

From a consensus process point of view, low distance-sum heterogeneity seems to be counterproductive due the structural characteristics this implies. In fig. 5.2.1 the left-hand extreme points are the networks *power grid* (fig. 5.2.2 a) and *Drosophila PIN* (fig. 5.2.2 b), and the right-hand extreme points are the networks *Skipwith* (fig. 5.2.2 c) and *Shelf* (fig. 5.2.2 d). If we review their S-plots (see figs. 5.2.3 and 5.2.4) we would expect the first two networks to present structural similarities (low distance-sum heterogeneities), and the last two would also be structurally similar (high distance-sum heterogeneities). Taking a close look to their structural characteristics, given in table 5.2.1, we can see that both *power grid* and *Drosophila PIN* show low average degree, large diameter, large average path length, low density and low algebraic connectivity, all with negative impact for

consensus, whereas the other two networks, *Skipwith* and *Shelf* show the opposite: high average degree, low diameter, low average path length, relative high density and high algebraic connectivity, all with positive impact for consensus.

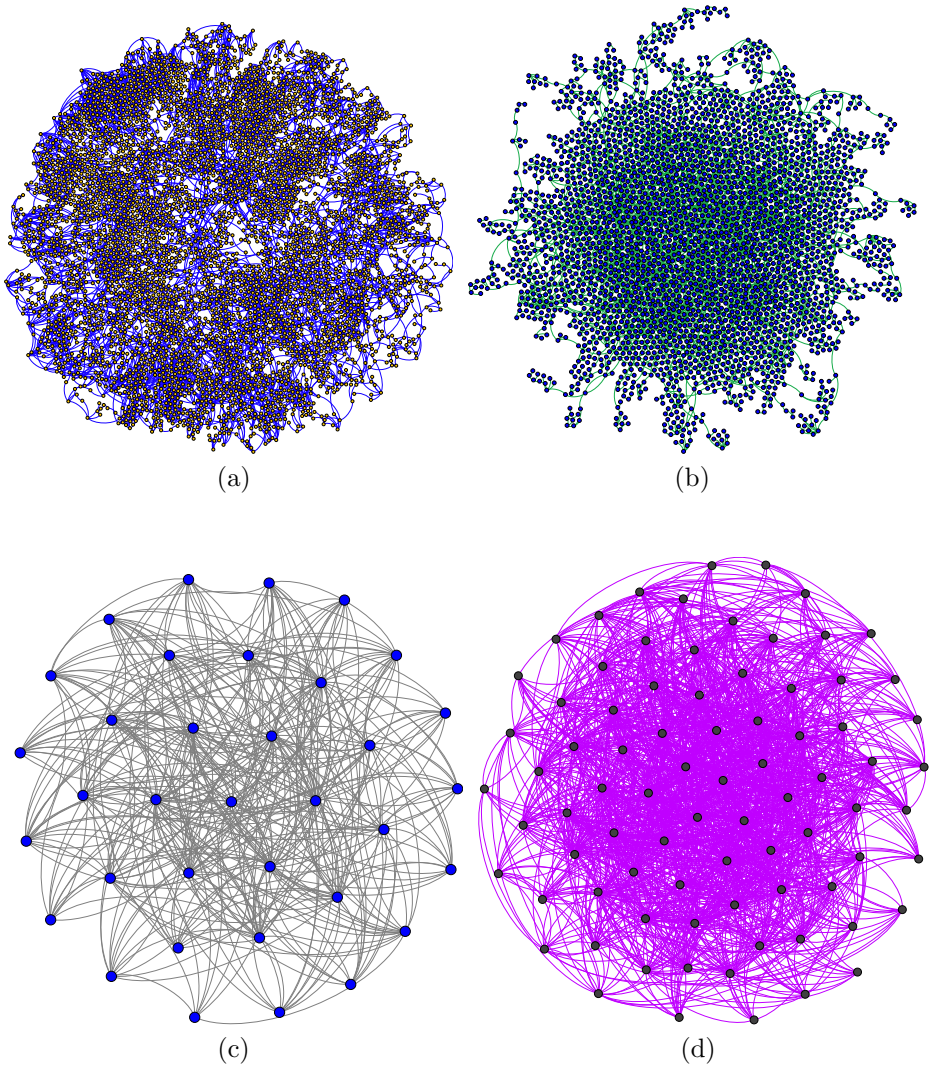
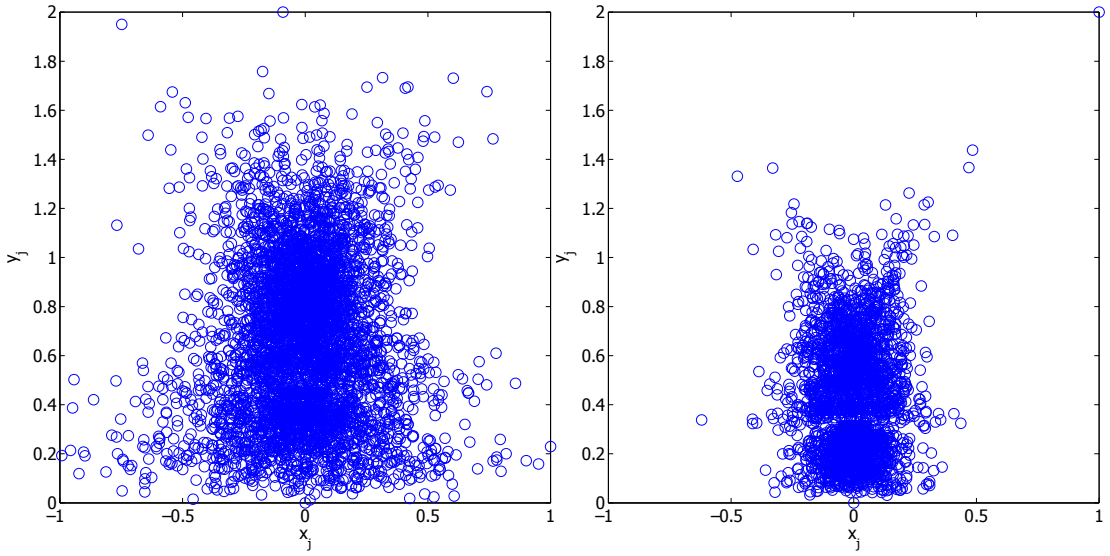


Figure 5.2.2: Illustration of networks: (a) power grid, (b) Drosophila PIN, (c) Skipwith and (d) Shelf

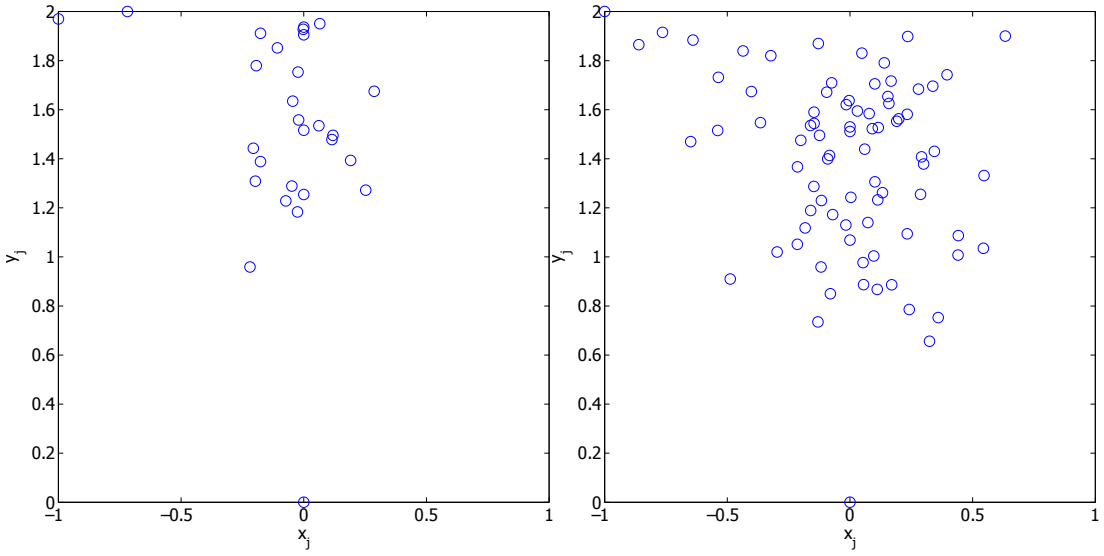
Network	average time for consensus	average degree	diameter	average path length	density	algebraic connectivity
power grid	276221.87	2.669	46	18.98	0.001	0.001
Drosophila-PIN	197708.37	2.408	27	9.43	0.001	0.001
Skipwith	85.18	20.171	3	1.417	0.593	7.604
Shelf	254.03	35.827	3	1.573	0.448	7.542

Table 5.2.1: Structural characteristics of the networks power grid, Drosophila-PIN, Skipwith and Shelf



(a) (b)

Figure 5.2.3: S-plots for the left-hand extreme networks: (a) power grid (b) Drosophila-PIN



(a) (b)

Figure 5.2.4: S-plots for the right-hand extreme networks: (a) Skipwith (b) Shelf

Summary

We have introduced a mathematical index as a way to overcome the difficulties that arise when trying to characterise complex networks according to their distance-sum distributions. This index, called distance-sum heterogeneity index, was applied to study this feature in different kinds of networks, and we provided some interesting results regarding the behavior of the index: in general all the networks studied presented low values of heterogeneity in distance-sum, and for some reason that is not clear, food webs showed the highest values of this index. We also proposed a visualisation of this characteristic by the use of the so-called S-plots, which makes use of spectral methods.

The results of the correlations between the time for consensus and some structural characteristics obtained in the previous chapter support those given in table 5.2.1. When analysing a consensus process in a network we can verify every single structural characteristic in order to stablish whether or not such network would make easier/harder reaching consensus. What is interesting from our results is that the distance-sum heterogeneity index induces an order that is closely related to these

structural characteristics, that we have showed are important for the dynamics of consensus: average degree, diameter, average path length, algebraic connectivity. Thus, we may use this distance-based parameter, together with the S-plots, to get a pretty good idea in the first instance about how hard/easy could be reaching consensus in a network regardless its nature. Of course, a more complete analysis would require further review of the structure of the network, but the distance-sum heterogeneity index can be useful for analysing consensus or other synchronisation processes in complex networks.

Chapter 6

CONCLUSIONS

We have presented a model to address the unexplored influence of the combined action of direct and indirect peer pressure on the dynamics of social groups. Our model considers that the consensus dynamics is controlled not only by the agreement between directly connected peers, but also by the influence of those peers which are socially or culturally close to them. The results obtained with this generalised consensus model highlight the important role played by the indirect peer pressure on the processes of consensus, leadership and diffusion of innovations in social groups.

Consensus is known to be influenced by a small group of actors playing a driving role, these actor are called leaders and can guide the behaviour of the entire network. The function of this drivers in the system controllability, an in particular their status or position in the complex network, has recently received great attention. As expected, the presence of leaders reduces significantly the time for reaching consensus in the network. In terms of controlling the system, our findings show that appropriate levels of indirect peer pressure allows that randomly emerging leaders could be as good as those occupying special positions or centrality in the network.

We have also studied two important factors that directly impact the controllability processes of a network. The first factor is the role played by the presence of tightly connected groups or communities of nodes in the system. The other one is the cohesiveness of the leaders trying to drive the consensus of the while network. In

both cases, our results show that if the level of indirect peer pressure is relatively weak, local leaders and leaders with strong cohesiveness are the best groups for controlling the network. However, as the indirect peer pressure increases, the barriers imposed by the communities and leaders' cohesiveness vanish, and the networks are easily controlled, even by leaders emerging from random positions.

Another area in which we have found indirect peer pressure plays an important role is in the diffusion of innovations process. In this case our results, based on real-world data coming from two studies on diffusions of innovations in different scenarios, show that a moderate indirect peer pressure is needed in order to reproduce the rates of diffusion of these innovations, independently of the social scenario in which they took place. Our findings offer a new perspective for the analysis of consensus process in social groups, and also place some important questions about the role that indirect peer pressure plays in controlling or driving social networks.

When analysing the impact of having tightly connected groups in a network, we have to consider two well known structural characteristics: the degree of nodes, and the distance among nodes. Different works have studied the number of connections a node has in a network, and have proposed metrics based on this value to characterise complex topologies. For the case of the distances among nodes, we studied the distance-sum heterogeneity of artificially constructed random networks, as well as real-world networks.

The distance-sum value forms part of different graph-theoretic invariants used for studying graphs and networks in different fields. Our work first provides an analysis of the distance-sum cumulative distributions, as a natural extension of what has been widely done for the degree of the nodes among the works on networks. In this part we have shown that these distance-sum distributions dot not account for the heterogeneity in the distance-sums of random and real-world networks. As a result of the last finding, we have proposed and introduced an index of distance-sum heterogeneity based on a hypothetical consensus process where all nodes look to agree on their distance-sums vales. Derived from this model, our index takes a quadratic form of the Laplacian matrix of the graph under analysis. This distance-sum heterogeneity index allows an interpretation of the Balaban index of a graph to be the contribution of the average time needed by n walkers to independently visiting their neighbours minus the contribution of the distance-sum heterogeneity

of the graph.

From a computational analysis, we conjecture that the maximum value of the distance-sum heterogeneity index for a graph with n nodes is reached for the so called complete split graph. Using the distance-sum heterogeneity values of the complete split graphs we proposed an index of relative distance-sum heterogeneity, and we have shown that this index differentiates random graphs with different degree distributions, as well as real-world networks from different nature and different topologies. Moreover, this new index can give good insights about how fast consensus can be reached in a particular network regardless its nature, this is due the distance-sum heterogeneity correlates good with different structural properties which are important for any consensus process.

Two lower bounds for the time for consensus were derived and tested for some networks, the results point to the idea that the higher bound t_{highB} might be more appropriated to bound the time for consensus for networks, although this is just a preliminary insight and further analysis must be carried out.

6.1 Future work

Our work opens opportunities for further analysis as future work: naturally, and derived from the kind of topologies explored in this thesis, the studies should continue with the inclusion of directed networks, as well as changing topologies, consensus processes with delays or faults in transmission of information. For the case of analysing the effects of indirect peer pressure on social networks, the study can be adapted and performed using other models for social dynamic opinion formation. The extension of the present work will take us to the ground of non-linear dynamics, which would enrich the recent findings.

The evidence provided highlights the importance of the information coming from the analysis of the relative distance-sum index, this additional information should be taken into account while carrying out structural analysis of networks. The complete split graph SK_{n,α^*} with n nodes has $m = \frac{(n-\alpha^*)(n+\alpha^*-1)}{2}$ links, however, most real-world networks have different (perhaps lower) densities than a complete split graph with the same number of nodes. Consequently, our future steps go in

the direction of finding the graphs that maximize the distance-sum heterogeneity index for a given number of nodes and a given number of links. It is worth it to continue walking on this avenue to propose methods and algorithms looking to solve questions like why food webs and possibly other ecological networks display larger distance-sum heterogeneity than other kind of networks. A further characterisation of the spectral representation of the distance-sum heterogeneity index should give more interesting insights about it. A good approach for analysing the influence of the distance-sum heterogeneity index in a consensus would be to consider a consensus process for a complete split graph $SK_{n,\alpha}$ and then rewire the links until we have a ER random network.

From our analysis we can see that two main structural characteristics play important roles in the dynamics of the leader-follower consensus when peer pressure is present: the degree and the total distance among nodes. We envision that a combined analysis of these characteristics, i. e. the "clumpiness" of a network, would provide valuable insights for the study of these kind of phenomena and will build on the open question about which nodes would be structurally the best drivers for controlling networks.

Finally, the analysis of the lower bounds derived in this work (see 3.1.9 and 3.1.8) shows that the bound given by 3.1.8 could be the best one when trying to known a priori the time for consensus, thus a more extensive analysis with a larger set of networks must be carried out to confirm this.

Appendix A

Theorems

A.1 Gershgoring's Theorem.

Let $A \in \mathbb{C}^{n\times n}$ and suppose that $X^{-1}AX = D+F$ where $D = diag(d_1, \ldots, d_n)$ and F has no nonzero diagonal entries. Then the eigenvalues of A lie in the union of the disks Δ_1, $\Delta_2, \ldots,$ Δ_n where

$$\Delta_i = \left\{ z \in \mathbb{C} : \ |z - d_i| \leq \sum_{j=1}^{n} |f_{ij}| \right\}.$$

If we choose X carefully we often get tight bounds on the locations of the eigenvalues. Simple choices of X can also be useful. Note that if the disks are disjoint they each contain a single eigenvalue of A.

A.2 Perron's theorem

Suppose $A \in \mathbb{R}^{n\times n}$ and $A > 0$. Let ρ and σ be respectively the spectral radius and the spectrum of A. Then A has an eigenvalue λ that satisfies the following properties.

1. $\lambda = \rho(A)$.

2. If $\mu \in \sigma(A)$ and $\mu \neq \lambda$ then $|\mu| < \lambda$.

3. λ has algebraic multiplicity 1.

4. If $Ay = \lambda y$ then $y = ax$ where $x > 0$ and $\alpha \in \mathbb{C}$.

A.3 Perron-Frobenius theorem

If A is fully indecomposable and nonnegative, then the properties listed in the Perron theorem still hold. If it is irreducible, then 1, 3 and 4 are guaranteed to hold.

Appendix B

Datasets description, tables and figures

B.1 Datasets description.Networks used for the different analysis in this work (see [5] and references therein).

Name	nodes	Type	Description
Benguela	29	ecological	Marine ecosystem of Bengela, off the south-west coast of South Africa.
Bridge Brook	75	ecological	Pelagic species from the largest of set of fifty Adirondack Lake (NY) food webs.
BF23, BF70, BF71	40, 48, 49	social	Networks of friendship ties from the communities identified as 23, 70, and 71 from the Brazilian Farmers longitudinal study on the adoption of a new corn seed.

Name	nodes	Type	Description
Canton	108	ecological	Primarily invertebrates and algae in a tributary, surrounded by pasture, of the Taieri River in the South Island of New Zealand.
Centrality-literature	118	informational	Citation network of papers published in the field of Network Centrality.
Chesapeake	33	ecological	The pelagic portion of an eastern US estuary, with an emphasis on larger fish.
Coachella	30	ecological	Wide range of highly aggregated taxa from the Coachella Valley desert in Southern California.
Colorado Springs	324	social	The risk network of persons with HIV infection during its early epidemic phase in Colorado Springs, USA, using analysis of community-wide HIV/AIDS contact tracing records (sexual partners) during 1985-99.
Corporate	1586	social	American corporate elite formed by the directors of the 625 largest corporations that reported the compositions of their boards, selected from the Fortune 1,000 in 1999.
Dolphins	62	ecological	Social network of a bottlenose dolphins (Tursiops truncates) population near New Zealand.
Drosophila PIN	3039	biological	Protein-protein interaction network in Drosophila melanogaster (fruit fly).
Drugs	616	social	Social network of injecting drug-users (IDUs) who have shared a needle in the last six months.

Name	nodes	Type	Description
Electronic1A	122	technological	Electronic sequential logic circuits parsed from the ISCAS89 benchmark set, where nodes represent logic gates and flip-flops.
ER	150	random	Simple undirected random graph generated from the Erdos-Renyi model implemented in the toolbox CONTEST.
BA	150	random	Simple undirected random graph generated from the preferential attachment model implemented in the toolbox CONTEST.
Electronic2A	252	technological	Electronic sequential logic circuits parsed from the ISCAS89 benchmark set, where nodes represent logic gates and flip-flops.
Electronic3A	512	technological	Electronic sequential logic circuits parsed from the ISCAS89 benchmark set, where nodes represent logic gates and flip-flops.
El Verde	156	ecological	Insects, spiders, birds, reptiles, and amphibians in a rainforest in Puerto Rico.
Galesburg	31	social	Friendship ties among 31 physicians.
GD	249	informational	Citation network of papers published in Proceedings of Graph Drawing during the period 1994-2000.
Geom	3621	social	Collaboration network of scientist in the field of computational gemoetry.
HighTech	33	social	Friendship ties among the employees in a small high-tech computer firm which sells, installs, and maintains computer systems.

Name	nodes	Type	Description
Hpyroli	710	biological	Protein-protein interaction network in H. pyroli.
HS	69	social	Heterosexual contacts, extracted at the Cadham Provincial Laboratory; a six-month block data from November 1997 to May 1998.
Internet 1997	3015	technological	The internet at the Autonomous System (AS) level, as of September 1997.
Internet 1998	3522	technological	The internet at the Autonomous System (AS) level, as of April 1998.
KSHV	50	biological	Protein-protein interaction network in Kaposi sarcoma herpes virus.
Little Rock	181	ecological	Pelagic and benthic species, particularly fish, zooplankton, microinvertebrates, and algae in Little Rock Lake, Wisconsin, USA.
Malaria PIN	229	biological	Protein-protein interaction network in P. falciparum (malaria parasite).
Math Method	30	social	This network concerns the diffusion of a new mathematics method in the 1950s. It traces the diffusion of the modern mathematical method among school systems that combine elementary and secondary programs in Allegheny County (Pennsylvania, USA.) .
Neurons	280	biological	Neuronal synaptic network of the nematode C. elegans.

Name	nodes	Type	Description
ODLIS	2998	informational	Vocabulary network of words related by their definitions in the Online Dictionary of Library and Information Science. Two words are connected if one is used in the definition of the other.
Afulgidus PIN	32	biological	Protein-protein interaction network in A. fulgidus.
Bsubtilis PIN	84	biological	Protein-protein interaction network in B. subtilis.
Ecoli PIN	230	biological	Protein-protein interaction network in E. coli.
Human PIN	2783	biological	Protein-protein interaction network in human.
Prison	67	social	Social network of inmates in prison who chose "Which fellows on the tier are you closest friends with?"
Reef Small	50	ecological	Caribbean coral reef ecosystem in Puerto Rico/Virgin Island shelf complex.
Roget	994	informational	Vocabulary network of words related by their definitions in Roget's Thesaurus of the English language. Two words are connected if one is used in the definition of the other.
Sawmill	36	social	Social communication network within a sawmill, where employees were asked to indicate the frequency with which they discussed work matters with each of their colleagues.

Name	nodes	Type	Description
Scotch Broom	154	ecological	Trophic interactions between the herbivores, parasitoids, predators, and pathogens associated with broom, Cytisus scoparius, collected in Silwood Park, Berkshire, England.
Skipwith	35	ecological	Invertebrates in an English pond.
Small World	233	informational	Citation network papers which cite Milgram's 1967 Psychology Today paper or include Small World in the title.
Social3	32	social	Social network among college students participating in a course about leadership. The students choose which three members they want to have on a committee.
Software Abi	1035	technological	Software network development for Abi.
Software Digital	150	technological	Software network development for Digital.
Software MySQL	1480	technological	Software network development for MySQL.
Software VTK	771	technological	Software network development for VTK.
Software XMMS	971	technological	Software network development for XMMS.
St. Marks	48	ecological	Mostly macroinvertebrates, fish, and birds associated with an estuarine seagrass community, Halodule wrightii, at the St. Marks Refuge, Florida, USA.
St. Martin	44	ecological	Birds and predators and arthropod prey of Anolis lizards on the island of St. Martin in the northern Lesser Antilles.

Name	nodes	Type	Description
Stony	112	ecological	Primarily invertebrates and algae in a tributary, surrounded by pasture, in native tussock habitat, of the Taieri River in the South Island of New Zealand.
Transc. Yeast	662	biological	Transcriptional regulation between genes in Saccaromyces cerevisiae.
Transc. Ecoli	328	biological	Transcriptional regulation between operons in Escherichia coli.
Transc. Urchin	45	biological	Developmental transcription network for sea urchin endomesoderm development.
USAir97	332	technological	Airport transportation network between airports in the US in 1997.
YeastS	2224	biological	Protein-protein interaction network in S. cerevisiae (yeast).
Ythan1	134	ecological	Mostly birds, fish, invertebrates, and metazona parasites in a Scottish estuary.
Ythan2	92	ecological	Reduced version of Ythan1, without parasites.
Zackary	34	social	Social network of friendship between members of the Zackary karate club.

B.2 Tables of simulation results

The times for consensus where calculated as the number of steps needed to reach consensus among the nodes of a network. The simulations follow the discrete time model for consensus with peer pressure 3.1.12. It was considered that a network had reached consensus when the between two consecutive measures of disagreement was less than or equal to the threshold 1e - 07. The initial states for all the nodes were randomly assigned with values between 0 and 1. Regarding the way the actors where allowed to interact, we considered 2 cases: 1) no peer pressure (No PP) which means that the interactions among actors where only with those directly connected, 2) peer-pressure. For the last case, the strength of the peer-pressure was changed according to the decays proposed in 3.1.2, 3.1.2 and 3.1.2, where actors were allowed to interact with neighbours beyond those directly connected. For each case, we considered 6 leaders, and averaged the results of 50 repetitions. The code used for calculating these times is given in Appendix C on page 200.

All the results for a particular network where summarized in tables divided into two main parts: at the top part of the table (*Random selection*) we present the average times for consensus corresponding to all the cases considered: *No PP*, *PL-decay* (with parameter α equal to 2 and 1.5), *Exp-decay* (with parameter β equal to 2 and 1.5), and *Social* (with parameter δ equal to 0.1, 0.025 and 0.5), and for all these cases, the node-leaders where randomly selected.

The second part of the table (below the *Random selection* part) has the heading *Centrality-based selection*, which contains the results of the simulations for the same decay cases as in the previous part, but the selection of the node-leaders was according to their centrality in the network (the centralities considered are those given in 3.2 on page 94), this means, those nodes with higher centrality were considered as leaders (six leaders for our case).

The values of time for consensus in each table are normalised to the highest value among all the cases (considering both, Random selection and Centrality-based selection) for comparison purposes, thus, the case of higher time for consensus shows value 1.

We also changed the parameter of divergence (as defined on page 96) among the positions (or states) of the leaders with respect to the average consensus value of the system, with values of 0.1 and 0.2 for all the networks, and values of 0, 0.1, 0.2 and 0.5 for the Sawmill social network (this network was used for illustrating the effects of PP when there is divergence among leaders).

B.2.1 Normalised consensus times for random networks and real networks for the study consensus under peer pressure (times were computed with a divergence value of 0.1)

- *BA* - random network

	Random selection							
	No PP	PL-decay		Exp-decay		Social		
		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
	1.00	0.44	0.31	0.75	0.67	0.84	0.38	0.13
	Centrality-based selection							
	No PP	PL-decay		Exp-decay		Social		
Centrality		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
BC	0.89	0.33	0.26	0.73	0.61	0.41	0.23	0.09
CC	0.91	0.34	0.27	0.78	0.65	0.42	0.22	0.08
DC	0.89	0.33	0.26	0.75	0.62	0.41	0.23	0.09
EC	0.93	0.34	0.27	0.77	0.64	0.42	0.22	0.08
SC	0.90	0.34	0.27	0.76	0.63	0.41	0.22	0.08

- *ER* - random network

	Random selection							
	No PP	PL-decay		Exp-decay		Social		
		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
	1.00	0.34	0.23	0.90	0.75	0.61	0.31	0.10
	Centrality-based selection							
	No PP	PL-decay		Exp-decay		Social		
Centrality		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
BC	0.81	0.10	0.06	0.67	0.47	0.28	0.10	0.05
CC	0.81	0.10	0.06	0.67	0.46	0.27	0.09	0.02
DC	0.81	0.10	0.06	0.69	0.48	0.28	0.10	0.03
EC	0.82	0.13	0.08	0.69	0.48	0.30	0.11	0.03
SC	0.82	0.13	0.08	0.68	0.48	0.31	0.11	0.02

- *Corporate* social network

	Random selection							
	No PP	PL-decay		Exp-decay		Social		
		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
	1.00	0.36	0.22	0.96	0.92	0.94	0.37	0.11
	Centrality-based selection							
	No PP	PL-decay		Exp-decay		Social		
Centrality		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
BC	0.67	0.02	0.01	0.52	0.35	0.27	0.08	0.01
CC	0.70	0.03	0.01	0.54	0.35	0.28	0.09	0.01
DC	0.69	0.05	0.04	0.52	0.36	0.29	0.10	0.02
EC	0.67	0.02	0.01	0.52	0.36	0.27	0.08	0.01
SC	0.68	0.02	0.01	0.51	0.35	0.27	0.08	0.01

- *Drugs* social network

	Random selection							
	No PP	PL-decay		Exp-decay		Social		
		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
	1.00	0.17	0.12	0.89	0.64	0.53	0.24	0.08
	Centrality-based selection							
	No PP	PL-decay		Exp-decay		Social		
Centrality		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
BC	0.97	0.07	0.04	0.79	0.51	0.40	0.12	0.02
CC	0.79	0.08	0.05	0.73	0.50	0.40	0.12	0.02
DC	0.77	0.04	0.06	0.66	0.45	0.36	0.13	0.06
EC	0.80	0.11	0.08	0.71	0.53	0.43	0.13	0.04
SC	0.80	0.11	0.08	0.72	0.50	0.42	0.13	0.04

- *Prison* social network

	Random selection							
	No PP	PL-decay		Exp-decay		Social		
		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
	1.00	0.26	0.19	0.93	0.70	0.49	0.23	0.08
	Centrality-based selection							
	No PP	PL-decay		Exp-decay		Social		
Centrality		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
BC	0.93	0.17	0.11	0.85	0.68	0.45	0.17	0.04
CC	0.97	0.17	0.11	0.86	0.66	0.45	0.17	0.04
DC	0.95	0.17	0.10	0.85	0.65	0.44	0.17	0.04
EC	0.98	0.17	0.11	0.84	0.68	0.44	0.17	0.04
SC	0.94	0.17	0.10	0.85	0.65	0.43	0.17	0.04

- *Zackary* social network

	Random selection							
	No PP	PL-decay		Exp-decay		Social		
		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
	1.00	0.27	0.19	0.94	0.67	0.39	0.19	0.08
	Centrality-based selection							
	No PP	PL-decay		Exp-decay		Social		
Centrality		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
BC	0.73	0.22	0.17	0.63	0.51	0.32	0.16	0.06
CC	0.75	0.22	0.16	0.66	0.50	0.32	0.15	0.06
DC	0.76	0.22	0.17	0.64	0.51	0.32	0.16	0.06
EC	0.75	0.21	0.17	0.64	0.50	0.33	0.16	0.05
SC	0.76	0.22	0.17	0.65	0.49	0.31	0.16	0.06

- *Colorado Springs* social network

	Random selection							
	No PP	PL-decay		Exp-decay		Social		
		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
	0.83	0.22	0.15	0.76	0.62	0.56	0.37	0.13
	Centrality-based selection							
	No PP	PL-decay		Exp-decay		Social		
Centrality		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
BC	1.00	0.11	0.07	0.87	0.69	0.53	0.25	0.04
CC	0.90	0.07	0.03	0.82	0.53	0.27	0.27	0.04
DC	0.89	0.13	0.09	0.82	0.65	0.50	0.24	0.06
EC	0.85	0.13	0.06	0.75	0.62	0.47	0.22	0.04
SC	0.95	0.17	0.12	0.86	0.69	0.54	0.28	0.07

- *Dolphins* social network

	Random selection							
	No PP	PL-decay		Exp-decay		Social		
		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
	0.82	0.25	0.16	0.76	0.61	0.38	0.20	0.08
	Centrality-based selection							
	No PP	PL-decay		Exp-decay		Social		
Centrality		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
BC	0.77	0.17	0.11	0.70	0.56	0.37	0.15	0.06
CC	0.92	0.17	0.12	0.65	0.64	0.43	0.18	0.05
DC	0.72	0.18	0.12	0.64	0.57	0.39	0.17	0.06
EC	0.96	0.21	0.14	0.99	0.79	0.52	0.21	0.06
SC	1.00	0.22	0.14	0.97	0.77	0.54	0.21	0.06

- *Galesburg* social network

	Random selection							
	No PP	PL-decay		Exp-decay		Social		
		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
	0.83	0.23	0.16	0.71	0.59	0.39	0.17	0.06
	Centrality-based selection							
	No PP	PL-decay		Exp-decay		Social		
Centrality		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
BC	0.81	0.21	0.14	0.73	0.59	0.37	0.16	0.05
CC	0.98	0.24	0.16	0.87	0.66	0.40	0.16	0.05
DC	0.98	0.23	0.16	0.84	0.64	0.41	0.17	0.05
EC	1.00	0.23	0.17	0.86	0.65	0.41	0.17	0.06
SC	1.00	0.24	0.17	0.87	0.68	0.41	0.17	0.06

- *HS* social network

	Random selection							
	No PP	PL-decay		Exp-decay		Social		
		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
	0.75	0.20	0.16	0.69	0.51	0.35	0.18	0.07
	Centrality-based selection							
	No PP	PL-decay		Exp-decay		Social		
Centrality		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
BC	0.65	0.18	0.14	0.56	0.43	0.28	0.15	0.06
CC	0.99	0.20	0.13	0.81	0.60	0.35	0.14	0.06
DC	0.69	0.17	0.11	0.59	0.46	0.27	0.13	0.07
EC	0.99	0.21	0.15	0.81	0.57	0.34	0.14	0.04
SC	1.00	0.21	0.15	0.78	0.59	0.34	0.14	0.04

- *High Tech* social network

	Random selection							
	No PP	PL-decay		Exp-decay		Social		
		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
	0.93	0.25	0.19	0.83	0.67	0.41	0.18	0.07
	Centrality-based selection							
	No PP	PL-decay		Exp-decay		Social		
Centrality		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
BC	0.78	0.22	0.16	0.70	0.56	0.37	0.16	0.06
CC	0.97	0.24	0.17	0.84	0.66	0.41	0.18	0.06
DC	0.93	0.24	0.18	0.83	0.67	0.40	0.18	0.06
EC	0.98	0.25	0.19	0.88	0.68	0.42	0.19	0.07
SC	1.00	0.27	0.19	0.88	0.68	0.44	0.19	0.06

- *Math Method* social network

	Random selection							
	No PP	PL-decay		Exp-decay		Social		
		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
	0.87	0.26	0.19	0.77	0.61	0.41	0.19	0.07
	Centrality-based selection							
	No PP	PL-decay		Exp-decay		Social		
Centrality		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
BC	0.63	0.22	0.16	0.59	0.50	0.35	0.17	0.06
CC	1.00	0.26	0.19	0.87	0.71	0.47	0.19	0.07
DC	0.98	0.26	0.18	0.86	0.70	0.46	0.19	0.07
EC	0.96	0.27	0.18	0.90	0.74	0.45	0.20	0.07
SC	1.00	0.26	0.19	0.86	0.71	0.44	0.20	0.07

- *Sawmill* social network

	Random selection							
	No PP	PL-decay		Exp-decay		Social		
		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
	1.00	0.24	0.17	0.88	0.68	0.45	0.20	0.07
	Centrality-based selection							
	No PP	PL-decay		Exp-decay		Social		
Centrality		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
BC	0.72	0.19	0.13	0.65	0.54	0.38	0.17	0.05
CC	0.75	0.19	0.12	0.70	0.56	0.40	0.17	0.05
DC	0.74	0.19	0.12	0.71	0.59	0.40	0.17	0.05
EC	0.99	0.22	0.15	0.92	0.68	0.45	0.18	0.05
SC	0.82	0.20	0.13	0.76	0.62	0.41	0.18	0.05

- *Social3* social network

	Random selection							
	No PP	PL-decay		Exp-decay		Social		
		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
	0.92	0.33	0.24	0.81	0.67	0.46	0.24	0.11
	Centrality-based selection							
	No PP	PL-decay		Exp-decay		Social		
Centrality		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
BC	0.92	0.31	0.24	0.79	0.68	0.42	0.21	0.07
CC	0.98	0.32	0.23	0.84	0.70	0.45	0.22	0.08
DC	1.00	0.33	0.25	0.87	0.70	0.45	0.23	0.08
EC	0.96	0.32	0.24	0.88	0.74	0.46	0.23	0.08
SC	0.92	0.32	0.25	0.86	0.69	0.46	0.22	0.09

B.2.2 Normalised consensus times for random networks and real networks for the study consensus under peer pressure (times were computed with a divergence value of 0.2)

- *BA* - random network

	Random selection							
	No PP	PL-decay		Exp-decay		Social		
		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
	1.00	0.42	0.29	0.94	0.85	0.79	0.33	0.11
	Centrality-based selection							
	No PP	PL-decay		Exp-decay		Social		
Centrality		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
BC	0.69	0.28	0.22	0.60	0.51	0.34	0.18	0.08
CC	0.74	0.29	0.22	0.64	0.54	0.35	0.18	0.08
DC	0.70	0.28	0.22	0.60	0.50	0.33	0.19	0.08
EC	0.72	0.28	0.23	0.63	0.53	0.34	0.18	0.07
SC	0.71	0.28	0.22	0.62	0.53	0.34	0.18	0.08

- *ER* - random network

	Random selection							
	No PP	PL-decay		Exp-decay		Social		
		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
	1.00	0.27	0.18	0.88	0.70	0.45	0.20	0.07
	Centrality-based selection							
	No PP	PL-decay		Exp-decay		Social		
Centrality		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
BC	0.67	0.21	0.16	0.57	0.46	0.33	0.18	0.07
CC	0.78	0.22	0.14	0.67	0.54	0.32	0.14	0.03
DC	0.70	0.22	0.18	0.58	0.45	0.36	0.20	0.08
EC	0.74	0.20	0.14	0.63	0.50	0.32	0.15	0.05
SC	0.75	0.23	0.17	0.64	0.51	0.36	0.18	0.07

- *Corporate* social network

	Random selection							
	No PP	PL-decay		Exp-decay		Social		
		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
	1.00	0.19	0.11	0.78	0.64	0.45	0.17	0.04
	Centrality-based selection							
	No PP	PL-decay		Exp-decay		Social		
Centrality		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
BC	0.22	0.01	0.00	0.17	0.11	0.09	0.03	0.002
CC	0.28	0.02	0.02	0.23	0.15	0.11	0.03	0.002
DC	0.24	0.04	0.03	0.20	0.13	0.12	0.04	0.007
EC	0.22	0.01	0.00	0.18	0.11	0.09	0.03	0.002
SC	0.23	0.01	0.00	0.18	0.11	0.09	0.03	0.002

- *Drugs* social network

	Random selection							
	No PP	PL-decay		Exp-decay		Social		
		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
	1.00	0.15	0.09	0.76	0.45	0.44	0.19	0.05
	Centrality-based selection							
	No PP	PL-decay		Exp-decay		Social		
Centrality		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
BC	0.69	0.06	0.04	0.52	0.34	0.25	0.06	0.02
CC	0.47	0.07	0.05	0.44	0.30	0.23	0.07	0.01
DC	0.46	0.04	0.05	0.37	0.24	0.20	0.08	0.04
EC	0.51	0.08	0.06	0.42	0.31	0.21	0.08	0.03
SC	0.51	0.08	0.06	0.40	0.32	0.21	0.08	0.03

- *Prison* social network

	Random selection							
	No PP	PL-decay		Exp-decay		Social		
		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
	1.00	0.29	0.20	0.91	0.73	0.50	0.24	0.08
	Centrality-based selection							
	No PP	PL-decay		Exp-decay		Social		
Centrality		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
BC	0.81	0.15	0.10	0.74	0.56	0.38	0.15	0.03
CC	0.83	0.15	0.09	0.73	0.57	0.39	0.15	0.04
DC	0.82	0.15	0.09	0.83	0.55	0.39	0.14	0.05
EC	0.90	0.15	0.10	0.79	0.58	0.39	0.15	0.03
SC	0.85	0.15	0.09	0.74	0.56	0.39	0.14	0.05

- *Zackary* social network

	Random selection							
	No PP	PL-decay		Exp-decay		Social		
		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
	1.00	0.27	0.21	0.83	0.64	0.40	0.19	0.08
	Centrality-based selection							
	No PP	PL-decay		Exp-decay		Social		
Centrality		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
BC	0.72	0.23	0.18	0.62	0.49	0.30	0.14	0.07
CC	0.72	0.23	0.18	0.61	0.48	0.30	0.14	0.06
DC	0.72	0.22	0.17	0.60	0.47	0.29	0.16	0.06
EC	0.70	0.22	0.17	0.59	0.48	0.31	0.15	0.05
SC	0.73	0.22	0.18	0.59	0.48	0.29	0.15	0.06

- *Colorado Springs* social network

	Random selection							
	No PP	PL-decay		Exp-decay		Social		
		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
	0.70	0.27	0.16	0.63	0.57	0.64	0.40	0.11
	Centrality-based selection							
	No PP	PL-decay		Exp-decay		Social		
Centrality		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
BC	1.00	0.11	0.07	0.86	0.63	0.50	0.20	0.04
CC	0.84	0.07	0.04	0.74	0.58	0.49	0.21	0.03
DC	0.81	0.13	0.09	0.73	0.56	0.43	0.20	0.05
EC	0.71	0.17	0.09	0.62	0.46	0.34	0.16	0.03
SC	0.91	0.17	0.11	0.85	0.64	0.49	0.24	0.06

- *Dolphins* social network

	Random selection							
	No PP	PL-decay		Exp-decay		Social		
		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
	0.84	0.24	0.18	0.76	0.59	0.41	0.19	0.08
	Centrality-based selection							
	No PP	PL-decay		Exp-decay		Social		
Centrality		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
BC	0.72	0.16	0.11	0.65	0.52	0.32	0.14	0.06
CC	0.92	0.16	0.10	0.82	0.60	0.40	0.14	0.05
DC	0.67	0.15	0.10	0.63	0.50	0.34	0.14	0.06
EC	0.94	0.19	0.12	0.84	0.63	0.41	0.17	0.05
SC	1.00	0.18	0.12	0.84	0.65	0.45	0.17	0.06

- *Galesburg* social network

	Random selection							
	No PP	PL-decay		Exp-decay		Social		
		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
	0.81	0.24	0.18	0.71	0.56	0.38	0.18	0.07
	Centrality-based selection							
	No PP	PL-decay		Exp-decay		Social		
Centrality		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
BC	0.80	0.20	0.15	0.70	0.56	0.36	0.15	0.06
CC	0.93	0.22	0.16	0.83	0.62	0.39	0.16	0.05
DC	0.94	0.22	0.16	0.83	0.64	0.39	0.16	0.06
EC	0.97	0.23	0.18	0.85	0.65	0.41	0.16	0.06
SC	1.00	0.24	0.17	0.84	0.64	0.40	0.17	0.06

- *HS* social network

	Random selection							
	No PP	PL-decay		Exp-decay		Social		
		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
	0.76	0.23	0.17	0.65	0.51	0.37	0.20	0.08
	Centrality-based selection							
	No PP	PL-decay		Exp-decay		Social		
Centrality		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
BC	0.60	0.20	0.16	0.51	0.39	0.27	0.16	0.07
CC	0.99	0.21	0.16	0.83	0.60	0.34	0.15	0.06
DC	0.66	0.14	0.11	0.53	0.43	0.25	0.13	0.07
EC	0.99	0.23	0.17	0.82	0.62	0.35	0.15	0.05
SC	1.00	0.24	0.17	0.81	0.60	0.34	0.15	0.05

- *High Tech* social network

	Random selection							
	No PP	PL-decay		Exp-decay		Social		
		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
	0.93	0.28	0.22	0.81	0.65	0.42	0.20	0.08
	Centrality-based selection							
	No PP	PL-decay		Exp-decay		Social		
Centrality		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
BC	0.76	0.22	0.17	0.69	0.56	0.35	0.16	0.06
CC	0.94	0.25	0.18	0.81	0.64	0.40	0.17	0.05
DC	0.95	0.24	0.19	0.82	0.66	0.40	0.17	0.07
EC	0.97	0.27	0.20	0.84	0.67	0.41	0.18	0.07
SC	1.00	0.27	0.20	0.86	0.69	0.42	0.18	0.06

- *Math Method* social network

	Random selection							
	No PP	PL-decay		Exp-decay		Social		
		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
	0.79	0.25	0.19	0.76	0.62	0.43	0.20	0.08
	Centrality-based selection							
	No PP	PL-decay		Exp-decay		Social		
Centrality		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
BC	0.60	0.23	0.18	0.56	0.49	0.32	0.16	0.06
CC	1.00	0.26	0.18	0.90	0.68	0.43	0.19	0.07
DC	0.98	0.25	0.18	0.87	0.69	0.42	0.19	0.07
EC	0.96	0.25	0.18	0.85	0.71	0.42	0.20	0.06
SC	0.93	0.25	0.18	0.85	0.70	0.43	0.19	0.06

- *Sawmill* social network

	Random selection							
	No PP	PL-decay		Exp-decay		Social		
		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
	0.96	0.23	0.16	0.87	0.66	0.43	0.19	0.06
	Centrality-based selection							
	No PP	PL-decay		Exp-decay		Social		
Centrality		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
BC	0.63	0.17	0.12	0.59	0.46	0.33	0.14	0.05
CC	0.66	0.16	0.12	0.58	0.49	0.33	0.14	0.04
DC	0.64	0.15	0.10	0.60	0.48	0.33	0.14	0.04
EC	1.00	0.19	0.13	0.85	0.63	0.37	0.15	0.05
SC	0.75	0.17	0.12	0.65	0.51	0.35	0.14	0.04

- *Social3* social network

	Random selection							
	No PP	PL-decay		Exp-decay		Social		
		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
	0.94	0.37	0.29	0.83	0.71	0.50	0.28	0.13
	Centrality-based selection							
	No PP	PL-decay		Exp-decay		Social		
Centrality		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
BC	0.90	0.35	0.27	0.83	0.70	0.43	0.23	0.08
CC	0.97	0.34	0.27	0.87	0.73	0.45	0.22	0.09
DC	0.90	0.36	0.27	0.87	0.72	0.45	0.22	0.08
EC	1.00	0.31	0.25	0.87	0.73	0.44	0.23	0.09
SC	0.94	0.33	0.24	0.83	0.70	0.44	0.21	0.11

B.2.3 Normalised times for consensus - for Sawmill network with six leaders with divergences of 0.0, 0.1, 0.2, and 0.5.

- No divergence

	Random selection							
	No PP	PL-decay		Exp-decay		Social		
		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
	1.00	0.22	0.15	0.88	0.72	0.49	0.20	0.06
	Centrality-based selection							
	No PP	PL-decay		Exp-decay		Social		
Centrality		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
BC	0.80	0.21	0.14	0.73	0.62	0.43	0.43	0.05
CC	0.84	0.21	0.14	0.77	0.65	0.45	0.45	0.05
DC	0.83	0.21	0.14	0.76	0.64	0.45	0.45	0.06
EC	0.96	0.23	0.15	0.87	0.73	0.50	0.50	0.06
SC	0.89	0.23	0.14	0.85	0.69	0.46	0.46	0.06

- Divergence 0.1

	Random selection							
	No PP	PL-decay		Exp-decay		Social		
		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
	1.00	0.24	0.17	0.88	0.68	0.45	0.20	0.07
	Centrality-based selection							
	No PP	PL-decay		Exp-decay		Social		
Centrality		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
BC	0.72	0.19	0.13	0.65	0.54	0.38	0.17	0.05
CC	0.75	0.19	0.12	0.70	0.56	0.40	0.17	0.05
DC	0.74	0.19	0.12	0.71	0.59	0.40	0.17	0.05
EC	0.99	0.22	0.15	0.92	0.68	0.45	0.18	0.05
SC	0.82	0.20	0.13	0.76	0.62	0.41	0.18	0.05

- Divergence 0.2

	Random selection							
	No PP	PL-decay		Exp-decay		Social		
		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
	0.96	0.23	0.16	0.87	0.66	0.43	0.19	0.06
	Centrality-based selection							
	No PP	PL-decay		Exp-decay		Social		
Centrality		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
BC	0.63	0.17	0.12	0.59	0.46	0.33	0.14	0.05
CC	0.66	0.16	0.12	0.58	0.49	0.33	0.14	0.04
DC	0.64	0.15	0.10	0.60	0.48	0.33	0.14	0.04
EC	1.00	0.19	0.13	0.85	0.63	0.37	0.15	0.05
SC	0.75	0.17	0.12	0.65	0.51	0.35	0.14	0.04

- Divergence 0.5

	Random selection							
	No PP	PL-decay		Exp-decay		Social		
		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
	1.00	0.23	0.16	0.87	0.67	0.40	0.17	0.06
	Centrality-based selection							
	No PP	PL-decay		Exp-decay		Social		
Centrality		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
BC	0.59	0.15	0.11	0.53	0.43	0.27	0.11	0.04
CC	0.60	0.14	0.11	0.52	0.41	0.27	0.11	0.04
DC	0.60	0.12	0.08	0.54	0.43	0.27	0.10	0.03
EC	0.99	0.17	0.12	0.83	0.59	0.35	0.13	0.04
SC	0.72	0.14	0.10	0.60	0.47	0.29	0.11	0.04

B.2.4 Normalised times for consensus for a random graph with 10 communities (10 leaders) with and without PP. Leaders selected from their global centrality values and from their local (community) centrality.

- Leaders selected from global centrality

	Random selection							
	No PP	PL-decay		Exp-decay		Social		
		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
	0.15	0.01	0.01	0.12	0.06	0.04	0.01	0.003
	Centrality-based selection							
	No PP	PL-decay		Exp-decay		Social		
Centrality		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
BC	0.26	0.01	0.01	0.18	0.09	0.06	0.01	0.003
CC	0.46	0.01	0.01	0.32	0.16	0.10	0.02	0.002
DC	0.44	0.01	0.01	0.32	0.16	0.10	0.02	0.003
EC	1.00	0.01	0.01	0.62	0.27	0.16	0.03	0.003
SC	0.59	0.01	0.01	0.34	0.15	0.09	0.02	0.004

- Leaders selected from local centrality

	Random selection							
	No PP	PL-decay		Exp-decay		Social		
		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
	1.00	0.06	0.04	0.77	0.41	0.27	0.08	0.02
	Centrality-based selection							
	No PP	PL-decay		Exp-decay		Social		
Centrality		$\alpha = 2$	$\alpha = 1.5$	$\beta = 2$	$\beta = 1.5$	$\delta = 0.1$	$\delta = 0.25$	$\delta = 0.5$
BC	0.54	0.07	0.04	0.42	0.28	0.17	0.06	0.02
CC	0.40	0.07	0.04	0.31	0.22	0.14	0.06	0.02
DC	0.30	0.07	0.05	0.26	0.20	0.13	0.07	0.02
EC	0.43	0.07	0.04	0.34	0.25	0.15	0.06	0.02
SC	0.29	0.07	0.04	0.25	0.19	0.12	0.06	0.02

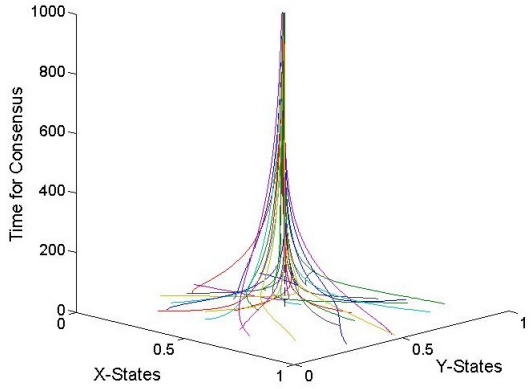
B.2.5 Distance-sum heterogeneity, and their correspondent relative values, for the networks studied in the *distance-sum heterogeneity* study.

Name	$\varphi(G)$	$\varphi_{rel}(G)$
Benguela	0.03230	9.646150
Bridge Brook	0.09850	11.451445
BF-23	0.00201	0.4375678
BF-70	0.00223	0.405278
BF-71	0.00232	0.4133079
Canton	0.01975	1.5984305
Centrality-literature	0.02940	2.1781903
Chesapeake	0.00595	1.5621755
Coachella	0.03660	10.552538
Colorado Springs	0.00050	0.0148677
Corporate	0.00395	0.0218380
Dolphins	0.00390	0.5479117
Drosophila-PIN	0.00030	0.0009246
Drugs	0.00275	0.0391287
Electronic1	0.00120	0.0853288
Electronic2	0.00085	0.0295300
Electronic3	0.00055	0.0094133
El Verde	0.03755	2.1049636
Galesburg	0.00720	2.008844
GD	0.00260	0.0914166
Geom	0.00285	0.0069486
High Tech	0.01285	3.3737739
Hpyroli	0.00390	0.0481461
Internet 97	0.00940	0.0273592
Internet 98	0.00940	0.0234145
HS	0.03350	0.6244568
KSHV	0.00935	1.6262342
Little Rock	0.07995	3.8637143
Malaria	0.00665	0.2542088

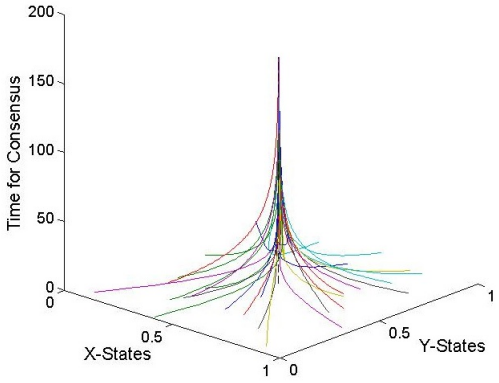
Name	$\varphi(G)$	$\varphi_{rel}(G)$
Math Method	0.00535	1.5425280
Neurons	0.0175	0.5332277
ODLIS	0.01575	0.0476017
Afulgidus-PIN	0.00455	1.2316740
Bsubtilis-PIN	0.00425	0.4418100
Ecoli-PIN	0.00640	0.2436192
Prison	0.00330	0.4296770
power grid	0.00004	0.0000751
Reef Small	0.05490	9.5486909
Roget	0.00230	0.0202852
Sawmill	0.00785	1.8935078
Scotch Broom	0.01455	0.8263308
Skipwith	0.12005	29.726216
Shelf	0.10315	11.108376
Small World	0.03865	1.4522468
Social3	0.00760	2.0573017
Abi-software	0.00280	0.0238358
Digital-software	0.00290	0.1690637
MySQL-software	0.00320	0.0190939
VTK-software	0.00370	0.0420647
XMMS-software	0.00130	0.0117373
St Marks	0.01450	2.6256156
St Martin	0.02360	4.2734157
Stony	0.02960	2.3086843
Yeast-transc	0.00225	0.0297890
Ecoli-transc	0.00335	0.0894588
Urchin-transc	0.00905	1.7472690
USAir97	0.03170	0.8363734
YeastS	0.00345	0.0136029
Ythan1	0.02535	1.6540355
Ythan2	0.02985	2.8320625
Zachary	0.01625	4.1387456

B.3 Figures

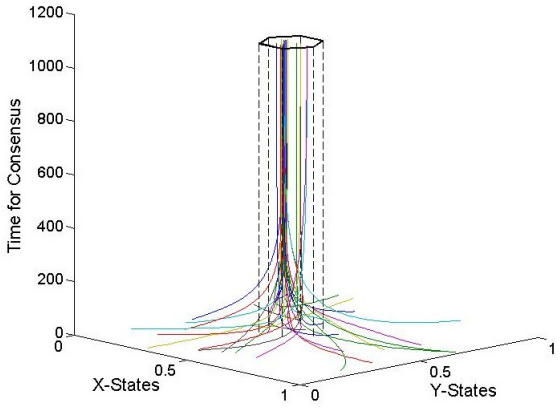
B.3.1 Trajectories for consensus processes with divergences 0.0, 0.1, 0.2 and 0.5 without PP and with PP (power-law decay with exponent 2) for the Sawmill social network with six leaders.



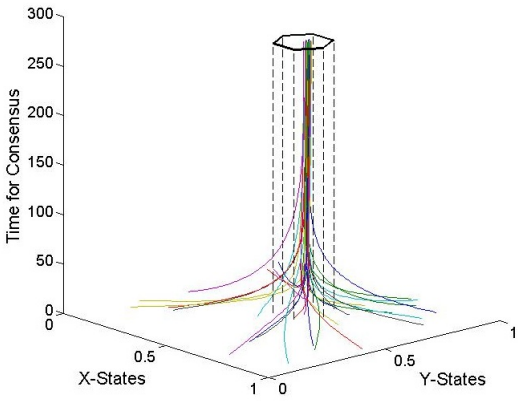
No PP, divergence = 0



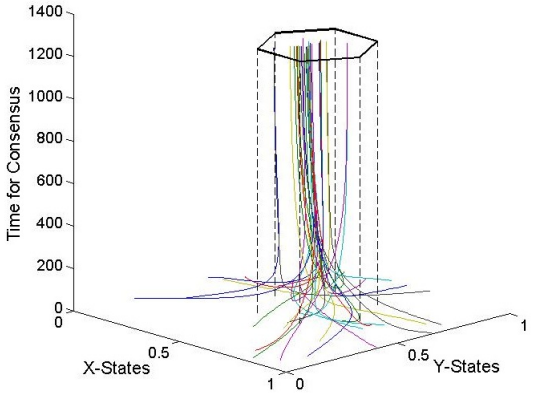
PP, divergence = 0



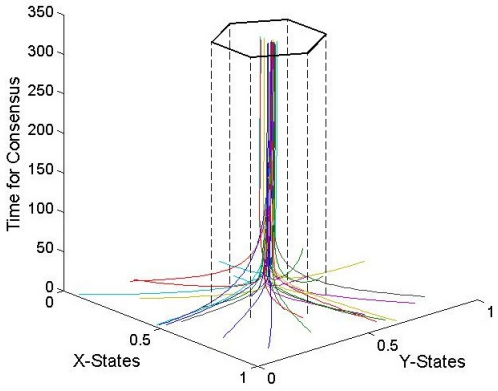
No PP, divergence = 0.1



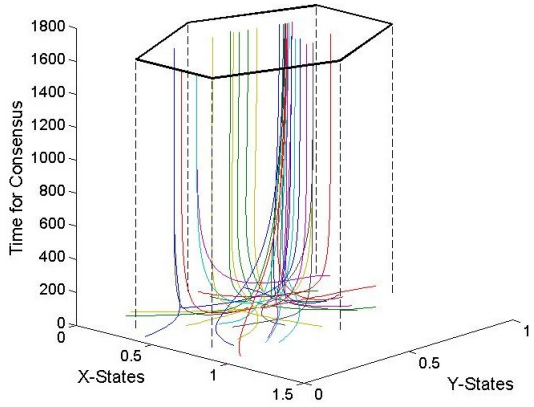
PP, divergence = 0.1



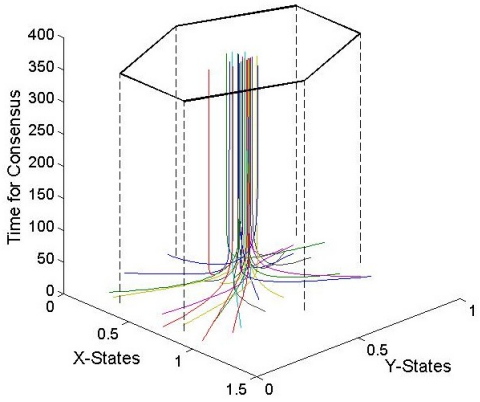
No PP, divergence = 0.2



PP, divergence = 0.2

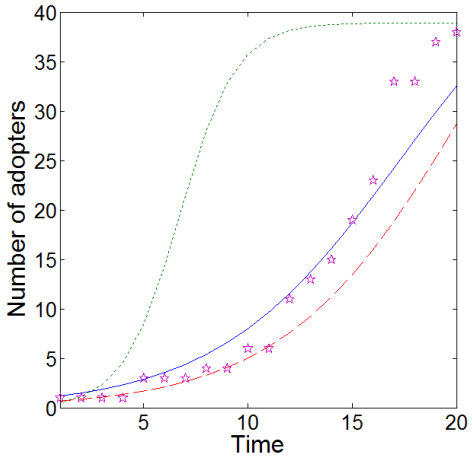


No PP, divergence = 0.5

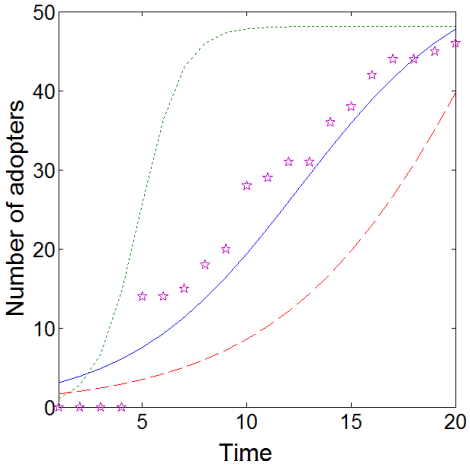


PP, divergence = 0.5

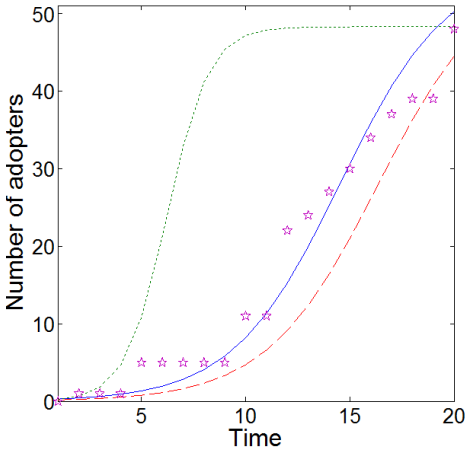
B.3.2 Diffusion curves for the four empirical social networks studied for the *consensus under peer pressure* study.



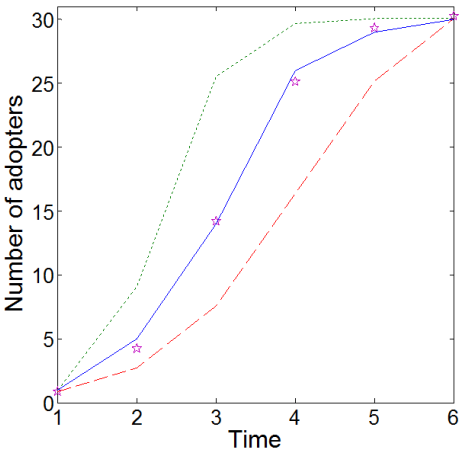
BF23



BF70

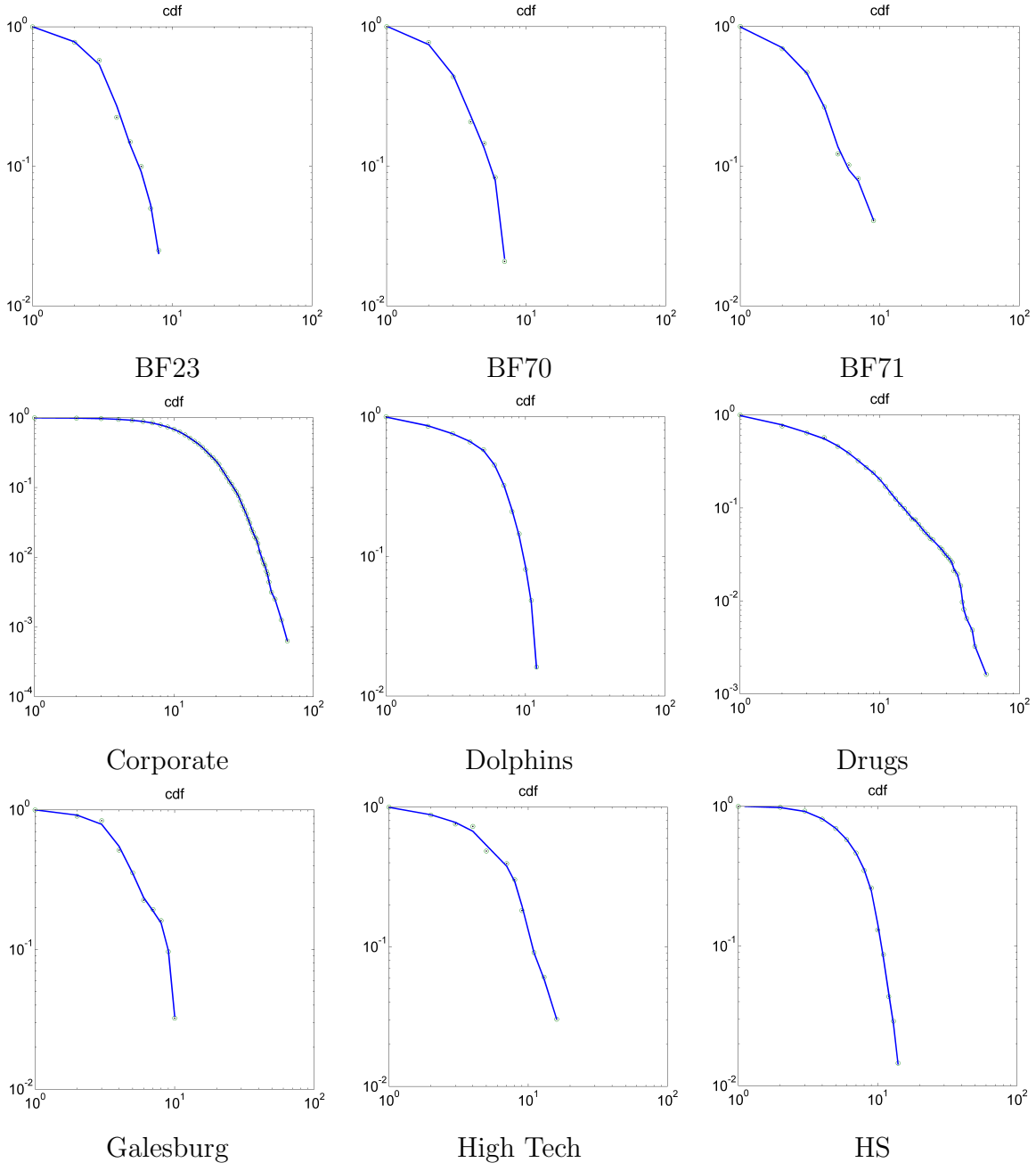


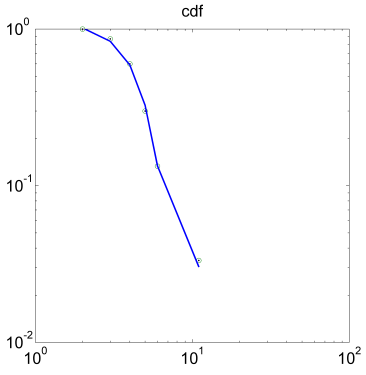
BF71



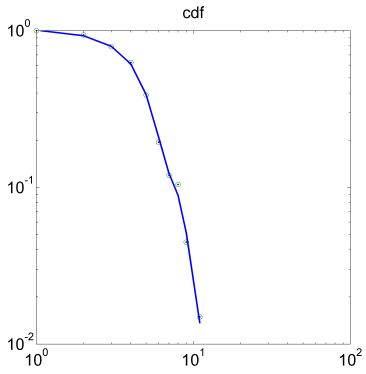
Math Method

B.3.3 Cumulative degree distribution for the set of 15 social networks studied for the *consensus under peer pressure* study.

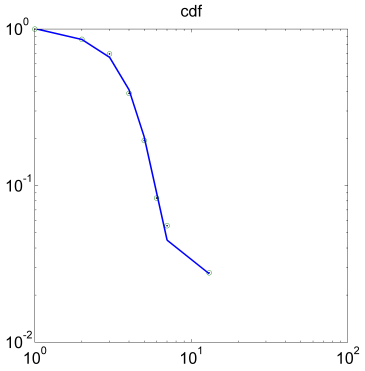




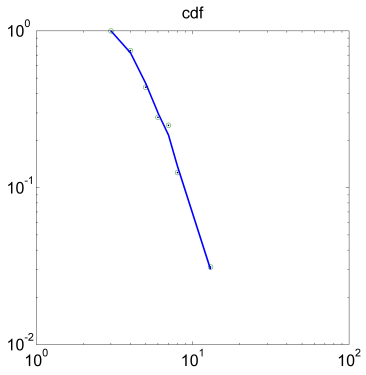
Math Method



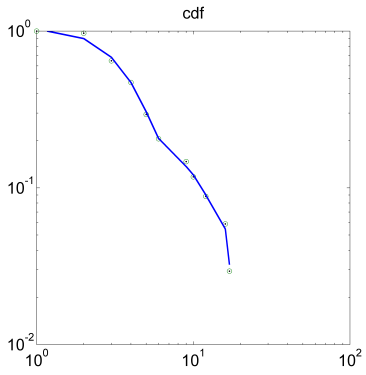
Prison



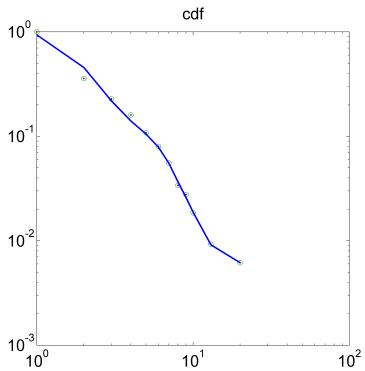
Sawmill



Social



Zackary



Colorado Springs

Appendix C

Matlab© scripts used for calculations

C.1 Script for calculating the distance-sum heterogeneity index $\varphi(G)$.

index.m

```
%% Distance-sum heterogeneity index
% This program computes the distance-sum heterogeneity
% index for all nodes in a given network
% with adjacency matrix A
function results=index(A);
A=A-diag(diag(A));
%Number of nodes
n=length(A);
%Identity matrix
I=eye(n,n);
%Matrix of distances
D=pathlength(A);
%Distance-sum vector
S=ones(1,n)*D;
```

```
S2= S.^(-1/2);
%Laplacian matrix
L=diag(ones(1,n)*A)-A;
%Distance-sum Heterogeneity Index
DSHI=0.5*(S2*L*S2');
results=DSHI';
```

C.2 Script for generating the S-plots to represent the distance-sum heterogeneity of a network.

spectral.m

```
%% Spectral representation of distance-sum
% This program generates  the S-plots for the spectral
% representations of the distance-sum heterogeneity of a
% given network with adjacency matrix A

% Set parameters
A=A-diag(diag(A));
n=length(A);
u=ones(1,n);
% Get Laplacian matrix
k=u*A;
K=diag(k);
L=K-A;
% Get Distance matrix and vector of distances
D=pathlength(A);
S=ones(1,n)*D;
S2= S.^(-1/2);
% Get spectral projections of values
[V,L]=eig(L);
angles=acosd(S.^(-0.5)*V/norm(S.^(-0.5)));
hipoth=real(diag(L).^0.5);
CA=cosd(angles).*hipoth';
CO=sind(angles).*hipoth';
% Normalise and plot values
```

```
nor=(sum(S.^-1)/(n-2*(n-1)^0.5))^0.5;
maxA=max(nor*CA);
minA=min(nor*CA);
divA=max(maxA,abs(minA));
maxO=max(nor*CO);
minO=min(nor*CO);
divO=max(maxO,abs(minO));
plot((nor*CA)/divA,(2*nor*CO)/divO,'o','MarkerSize',10);
xlabel('x_j');
ylabel('y_j');
axis square;
axis ([-1 1 0 2]);
xmean=mean((nor*CA)/divA);
ymean=mean((2*nor*CO)/divO);
xstd=std((nor*CA)/divA);
ystd=std((2*nor*CO)/divO);
centre=[xmean ymean];
x_length=2*xstd;
y_length=2*ystd;
max=max(x_length,y_length);
min=min(x_length,y_length);
ratio=min/max;
```

C.3 Script for calculating basic average time for consensus for a given network with and without peer pressure.

consensus.m

```
%% Time for average consensus
% Calculate the consensus with/without peer pressure
% Allows leader-nodes selection
% Generates random initial labels for leaders and
% random initial states for every process and repeats
% the simulations to get average values
%Inputs:
% -Name of file to be imported which contains
```

```
% the adjacency matrix of the network (must be in the
% same folder)
% -dismax: maximum difference between agent's states
% -Epsilon value: value of the parameter epsilon to
% be used with the stochastic matrix P
% -k Factor: the number of k-range interactions allowed
clear all;
tic;
clc;
%Loop to open the required file of Adj-matrix
fid = -1;
msg = '';
while fid < 0
   disp(msg);
   filename = input('Open file "name.extension":\n ', 's');
   fid,msg] = fopen(filename);
end
clear fid msg;

%Asign the imported file to the variable Matrix "A"
Aoriginal=load(filename);
%Eliminate any self-loops to make it simple graph
A=Aoriginal-diag(diag(Aoriginal));
%Size of matrix A (number of nodes 'n')
n=length(A);
clear filename;

%Input the maximum disagreement factor
dismax=input('disagreement max: ');
%Input the number of loops for calculating average consensus
num_loops=input('Number of loops: ');

%Calculate Distance matrix(A)
DM=pathlength(A);   %Use the function "pathlength"
kmax=max(max(DM));
%Calculate the diameter of A
str = fprintf('The graph has diameter= %d \n',kmax);
%Input the value of k factor
k=input('k-neighbours factor to be considered: ');

%Get vector of k-Laplacians and deltas_max
L=[];
```

```
for q=1:kmax
    [KLM,~,delta_max]=klp(n,q,DM);  %use function "klp"
    L(:,:,q)=KLM;   %Vector of k-Laplacians matrices
    deltas(q,1)=delta_max;
end
clear q KLM DM A delta_max;

KLMsum=L(:,:,1);
sumEps=deltas(1,1);

if k>1
%Here select kind of decay
INTERACTION=input('Interaction to be considered (100=PW 200=EXP
  300=SOC) :');
switch INTERACTION
    case 100
%CHOICE_1
%Calculate summation for Power-lay decay
%Input the value of exponent alfa>0
alfa=input('value of alfa (>0) to be considered: ');
for p=2:k
   KLMsum=(p.^(-alfa)).*(L(:,:,p))+KLMsum;
  sumEps=deltas(p,1)+sumEps;
end
Xfactor=0;
clear p;
    case 200
%CHOICE_2
%Calculate summation for Exponential decay
%Input the value of exponent alfa>0
alfa=input('value of alfa (>0) to be considered: ');
for p=2:k
    KLMsum=(exp(-alfa*p)).*(L(:,:,p))+KLMsum;
    sumEps=deltas(p,1)+sumEps;
end
Xfactor=0;
clear p;
    case 300
%CHOICE_3
%Calculate summation for Social interactions
%Input the value of X factor
Xfactor=input('Value of x factor: ');
```

```
92  for p=2:k
93      KLMsum=(p.*(Xfactor^(p-1))).*L(:,:,p)+KLMsum;
94      sumEps=deltas(p,1)+sumEps;
95  end
96  alfa=0;
97  clear p;
98      otherwise
99  warning('Unexpected value...')
100 end
101 else
102     INTERACTION=0;
103     Xfactor=0;
104     alfa=0;
105 end
106 clear L deltas;
107
108 %Value of Epsilon
109 str=fprintf('Epsilon value between 0 and %.7f:\n',(1/(sumEps)));
110 eps=input(':');
111 %Stochastic matrix P
112 P=eye(n)-(eps.*(KLMsum));
113
114 variable=1;
115 %Start loop for realizations
116 while variable < num_loops+1
117
118 %Random Vector of Initial X-states
119 x0=(rand(1,n));
120 xti=x0;
121
122 %Disagreement vector
123 Xdisagreement_vector=[];
124 disx=(xti*KLMsum*xti');
125 Xdisagreement_vector(1,:)=disx;
126 xt=0; yt=0;
127 x(1,:)=x0;
128
129 %Calculate consensus as x(k+1)=P*x(k)
130 while ((disx)>dismax)
131     xtf=(P*xti');   %Vector of X-states at time 't>0'
132     xti=xtf';
133     xt=xt+1;
```

```
134     tim(1,xt+1)=xt;
135     disx=(xtf'*KLMsum*xtf);
136     x(xt+1,:)=xtf';
137 
138 %Optional: Plot in real time every step
139  clf;
140  h=plot(tim,x);
141  axis square;
142  xlabel('time(s)');
143  ylabel('states');
144  title('Agreement')
145  drawnow;
146  hold off;
147  pause(.09)
148 
149 end
150 
151 Xdisagreement_vector(xt,:)=disx;
152 if disx>0
153 FINALX_dis(variable,1)=disx;
154 TIME_for_CONSENSUS(variable,1)=xt-1;
155 X0(variable,:)=x0;
156 X_states(:,:,variable)=x;
157 clc;
158 ciclo=variable;
159 variable=variable+1;
160 else
161 end
162 
163 end
164 
165 MEANtime=mean(TIME_for_CONSENSUS);
166 str = fprintf('Time for consensus: %.4f s\n',MEANtime);
167 str = fprintf('Interaction: %i \n',INTERACTION);
168 str = fprintf('k: %i \n',k);
169 str = fprintf('alpha: %f \n',alfa);
170 str = fprintf('x: %f \n',Xfactor);
171 toc
```

References

[1] D'Agostino, G. and Scala, A. *Networks of Networks: The Last Frontier of Complexity*. Springer, (2014).

[2] Meyers, R. A., editor. *Encyclopedia of Complexity and Systems Science*. Springer, (2009).

[3] Mesbahi, M. and Egerstedt, M. *Graph theoretic methods in multiagent networks*. Princeton University Press, (2010).

[4] Newman, M. *Networks: an introduction*. Oxford University Press, (2010).

[5] Estrada, E. *The structure of complex networks: theory and applications*. Oxford University Press, (2011).

[6] Kocarev, L. *Consensus and Synchronization in Complex Networks*. Springer, (2013).

[7] Olfati-Saber, R., Fax, J. A., and Murray, R. M. *Proceedings of the IEEE* **95**(1), 215–233 January (2007).

[8] Ren, W., Beard, R., and Atkins, E. In *American Control Conference, 2005. Proceedings of the 2005*, 1859–1864 vol. 3, June (2005).

[9] Dyer, J. R., Ioannou, C. C., Morrell, L. J., Croft, D. P., Couzin, I. D., Waters, D. A., and Krause, J. *Animal Behaviour* **75**(2), 461–470 February (2008).

[10] Sueur, C., Deneubourg, J.-L., and Petit, O. *PLoS ONE* **7**(2), e32566 February (2012).

[11] Vilone, D., Ramasco, J. J., Sánchez, A., and Miguel, M. S. *Scientific Reports* **2** September (2012).

[12] Fax, J. and Murray, R. *Automatic Control, IEEE Transactions on* **49**(9), 1465–1476 September (2004).

[13] Ren, W. and Beard, R. *Distributed consensus in multi-vehicle cooperative control: theory and applications*. Springer, (2007).

[14] Liu, Y.-Y., Slotine, J.-J., and Barabási, A.-L. *Nature* **473**(7346), 167–173 May (2011).

[15] Olfati Saber, R. and Murray, R. M. In *Proceedings of the 2003 American Controls Conference*, (2003).

[16] Ji, M. and Egersted, M. In *American Control Conference*, 4588, (2007).

[17] Haque, M. A., Egerstedt, M., and Martin, C. F. In *American Control Conference, 2008*, 3798–3803, (2008).

[18] Franceschelli, M., Martini, S., Egerstedt, M., Bicchi, A., and Giua, A. In *Decision and Control (CDC), 2010 49th IEEE Conference on*, 5775–5780, (2010).

[19] Garin, F. and Schenato, L. In *Networked Control Systems*, 75–107. Springer (2010).

[20] de la Croix, J.-P. and Egerstedt, M. In *2012 AAAI Fall Symposium Series*, (2012).

[21] Estrada, E., Kalala-Mutombo, F., and Valverde-Colmeiro, A. *Physical Review E* **84**(3) September (2011).

[22] Kalala-Mutombo, F. *Long-range Interactions in Complex Networks*. PhD thesis, University of Strathclyde, Glasgow, U.K., (2012).

[23] Aldous, J. M. and Wilson, R. J. *Graphs and applications: an introductory approach*, volume 1. Springer, (2000).

[24] DeGroot, M. H. *Journal of the American Statistical Association* **69**(345), 118–121 (1974).

[25] Dyer, J. R., Johansson, A., Helbing, D., Couzin, I. D., and Krause, J. *Philosophical Transactions of the Royal Society B: Biological Sciences* **364**(1518), 781–789 March (2009).

[26] Gatewood, J. B. *Topics in Cognitive Science* **4**(3), 362–371 July (2012).

[27] Golub, B. and Jackson, M. O. *Review of Network Economics, forthcoming* (2012).

[28] Groeber, P., Schweitzer, F., and Press, K. *arXiv preprint arXiv:0904.0761* (2009).

[29] Ianni, A. and Corradi, V. Discussion paper, April (2000).

[30] Notarstefano, G., Egerstedt, M., and Haque, M. In *Decision and Control, 2009 held jointly with the 2009 28th Chinese Control Conference. CDC/CCC 2009. Proceedings of the 48th IEEE Conference on*, 3733–3738, (2009).

[31] Berge, C. *The theory of graphs*. Courier Dover Publications, (2001).

[32] Wallis, W. D. *A beginner's guide to graph theory*. Springer, (2000).

[33] Newman, M. E. *Physical review letters* **89**(20), 208701 (2002).

[34] Watts, D. J. and Strogatz, S. H. *nature* **393**(6684), 440–442 (1998).

[35] van Steen, M. *Graph Theory and Complex Networks: An Introduction*. Maarten van Steen, (2010).

[36] Newman, M. E. *Proceedings of the National Academy of Sciences* **98**(2), 404–409 (2001).

[37] Merris, R. *Linear algebra and its applications* **197**, 143–176 (1994).

[38] Kac, M. *American Mathematical Monthly* , 1–23 (1966).

[39] Estrada, E. *Linear Algebra and its Applications* **436**(9), 3373–3391 May (2012).

[40] Estrada, E. and Knight, P. *A first course on complex networks*. Oxford University Press, (2015).

[41] Fiedler, M. *Czechoslovak Mathematical Journal* **23**(2), 298–305 (1973).

[42] Polderman, J. W. and Willems, J. C. *Introduction to mathematical systems theory: a behavioral approach*. Number 26. Springer, (1998).

[43] O'Shea, D. *An Introduction to Dynamical Systems and Mathematical Modelling*. State University of New York. Research Foundation, (1992).

[44] Rugh, W. J. *Linear system theory*. Prentice-Hall, Inc., (1996).

[45] Chapra, S. C. and Canale, R. P. *Numerical methods for engineers*, volume 2. McGraw-Hill, (2012).

[46] Barrat, A., Barthelemy, M., and Vespignani, A. *Dynamical processes on complex networks*, volume 1. Cambridge University Press Cambridge, (2008).

[47] Boccaletti, S., Latora, V., Moreno, Y., Chavez, M., and Hwang, D.-U. *Physics reports* **424**(4), 175–308 (2006).

[48] Pastor-Satorras, R. and Vespignani, A. *Physical review letters* **86**(14), 3200 (2001).

[49] Pecora, L. M. and Carroll, T. L. *Physical Review Letters* **80**(10), 2109 (1998).

[50] Axelrod, R. *Journal of conflict resolution* **41**(2), 203–226 (1997).

[51] Liggett, T. M. et al. *The Annals of Probability* **25**(1), 1–29 (1997).

[52] Guerra, B., Poncela, J., Gómez-Gardeñes, J., Latora, V., and Moreno, Y. *Physical Review E* **81**(5), 056105 (2010).

[53] Suchecki, K., Eguíluz, V. M., and San Miguel, M. *Physical Review E* **72**(3), 036132 (2005).

[54] Horn, R. A. and Johnson, C. R. *Matrix analysis*. Cambridge university press, (2012).

[55] Tanner, H. G., Pappas, G. J., and Kumar, V. *Robotics and Automation, IEEE Transactions on* **20**(3), 443–455 (2004).

[56] Tanner, H. G. In *Decision and Control, 2004. CDC. 43rd IEEE Conference on*, volume 3, 2467–2472. IEEE, (2004).

[57] Björkenstam, S., Ji, M., Egerstedt, M. B., and Martin, C. F. (2006).

[58] Borsche, T. and Attia, S. A. In *Control and Decision Conference (CCDC), 2010 Chinese*, 102, (2010).

[59] Ji, M., Muhammad, A., and Egerstedt, M. In *American control conference*, 1358–1363, (2006).

[60] Ferrari-Trecate, G., Egerstedt, M., Buffa, A., and Ji, M. In *Hybrid Systems: Computation and Control*, 212–226. Springer (2006).

[61] Lynch, N. A. *Distributed algorithms*. Morgan Kaufmann Publishers, San Francisco, Calif., (1996).

[62] Guerraoui, R. and Rodrigues, L. *Reliable Distributed Programming*, volume 138. Springer, (2006).

[63] Joyner, D., Van Nguyen, M., and Cohen, N. *Google Code* (2010).

[64] Lamport, L., Shostak, R., and Pease, M. *ACM Transactions on Programming Languages and Systems (TOPLAS)* **4**(3), 382–401 (1982).

[65] Franklin, S. and Graesser, A. *Lecture Notes in Computer Science* **1193**(3), 21–35 (1997).

[66] Davis, R. and Smith, R. G. *Artificial intelligence* **20**(1), 63–109 (1983).

[67] Vidal, J. M. *http://www.multiagent.com/fmas* **1** (2007).

[68] Shamma, J. S. *Cooperative control of distributed multi-agent systems*. Wiley Online Library, (2007).

[69] Reynolds, C. W. *SIGGRAPH Comput. Graph.* **21**(4), 25–34 August (1987).

[70] Vicsek, T. *Physical Review Letters* **75**(6), 1226–1229 (1995).

[71] Fax, J. A. *Optimal and cooperative control of vehicle formations*. PhD thesis, California Institute of Technology, (2001).

[72] Jadbabaie, A., Lin, J., and Morse, A. S. *Automatic Control, IEEE Transactions on* **48**(6), 988–1001 (2003).

[73] Mesbahi, M. In *American Control Conference, 2002. Proceedings of the 2002*, volume 2, 1234–1239. IEEE, (2002).

[74] Olfati-Saber, R. *CaltechCDSTR* (2003).

[75] Fax, J. A. and Murray, R. M. *CaltechCDSTR* (2001).

[76] Desai, J. P. *Journal of Robotic Systems* **19**(11), 511–525 (2002).

[77] Kuramoto, Y. and Araki, H. (1975).

[78] Zavlanos, M. M., Jadbabaie, A., and Pappas, G. J. In *46th Conference on Decision and Control*, 2919, (2007).

[79] Li, C.-T. and Lin, S.-D. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, 765, (2011).

[80] Tanner, H. G., Jadbabaie, A., and Pappas, G. J. *Automatic Control, IEEE Transactions on* **52**(5), 863–868 (2007).

[81] Olfati-Saber, R. *Automatic Control, IEEE Transactions on* **51**(3), 401–420 (2006).

[82] IEEE. *Average consensus over small world networks: A probabilistic framework*, (2008).

[83] Hovareshti, P. and Baras, J. S. In *Unifying Themes in Complex Systems*, 98–105. Springer (2008).

[84] IEEE. *Small world phenomenon, rapidly mixing Markov chains, and average consensus algorithms*, (2007).

[85] Wang, X. F. and Chen, G. *International Journal of Bifurcation and Chaos* **12**(01), 187–192 (2002).

[86] Olfati-Saber, R. In *American Control Conference, 2005. Proceedings of the 2005*, 2371–2378. IEEE, (2005).

[87] Dimarogonas, D. V. and Kyriakopoulos, K. J. *Automatic Control, IEEE Transactions on* **52**(5), 916–922 (2007).

[88] Lin, Z. et al. *Chinese Journal of Aeronautics* **28**(1), 191–199 (2015).

[89] AIAA Reston, VA. *Trajectory planning for coordinated rendezvous of unmanned air vehicles*, volume 4369, (2000).

[90] ACM. *Rendezvous: An architecture for synchronous multi-user applications*, (1990).

[91] Smith, S. L. *Strategies for rendezvous and formation stabilization of multi-agent systems*. PhD thesis, University of Toronto, (2005).

[92] Cortés, J., Martínez, S., and Bullo, F. *Automatic Control, IEEE Transactions on* **51**(8), 1289–1298 (2006).

[93] Lin, J., Morse, A. S., and Anderson, B. D. *SIAM Journal on Control and Optimization* **46**(6), 2096–2119 (2007).

[94] Pereira, S. S. *Distributed Consensus Algorithms for Wireless Sensor Networks*. PhD thesis, Ph. D. dissertation, Universitat Politecnica de Catalunya-Barcelona Tech, To be published, (2011).

[95] IEEE. *A scheme for robust distributed sensor fusion based on average consensus*, (2005).

[96] Lesser, V., Ortiz Jr, C. L., and Tambe, M. *Distributed sensor networks: A multiagent perspective*, volume 9. Springer Science & Business Media, (2012).

[97] Dhillon, S. S. and Chakrabarty, K. *Sensor placement for effective coverage and surveillance in distributed sensor networks*, volume 3. IEEE, (2003).

[98] IEEE. *Distributed multi-robot task assignment and formation control*, (2008).

[99] Gu, D. and Hu, H. *International Journal of Systems Science* **40**(5), 539–552 (2009).

[100] Ji, M. and Egerstedt, M. B. (2006).

[101] Lin, Z., Wang, L., Han, Z., and Fu, M. *Automatic Control, IEEE Transactions on* **59**(7), 1765–1777 (2014).

[102] IEEE. *Graph rigidity and distributed formation stabilization of multi-vehicle systems*, volume 3, (2002).

[103] Su, H. and Wang, X. *Pinning control of complex networked systems: Synchronization, consensus and flocking of networked systems via pinning.* Springer, (2013).

[104] Lin, C.-T. *Automatic Control, IEEE Transactions on* **19**(3), 201–208 (1974).

[105] Cao, Y. and Ren, W. In *Decision and Control, 2009 held jointly with the 2009 28th Chinese Control Conference. CDC/CCC 2009. Proceedings of the 48th IEEE Conference on*, 3014, (2009).

[106] Di Cairano, S., Pasini, A., Bemporad, A., and Murray, R. M. In *American Control Conference, 2008*, 1362, (2008).

[107] Dong, W. and Guo, Y. In *Cooperative Systems*, 369–386. Springer (2007).

[108] Jiang, F., Wang, L., Xie, G., Ji, Z., and Jia, Y. In *American Control Conference, 2009. ACC'09.*, 5665, (2009).

[109] Jin, Z. *Coordinated control for networked multi-agent systems.* PhD thesis, California Institute of Technology, (2007).

[110] Lozano, R., Spong, M. W., Guerrero, J. A., and Chopra, N. In *Decision and Control, 2008. CDC 2008. 47th IEEE Conference on*, 3713, (2008).

[111] Rahmani, A., Ji, M., Mesbahi, M., and Egerstedt, M. *SIAM Journal on Control and Optimization* **48**(1), 162–186 January (2009).

[112] Sontag, E. D. In *Mathematical system theory*, 453–462. Springer (1991).

[113] Zengqiang, C., Linying, X., and Zhuzhi, Y. In *Control Conference, 2008. CCC 2008. 27th Chinese*, 494, (2008).

[114] Banerjee, S. J. and Roy, S. *arXiv preprint arXiv:1209.3737* (2012).

[115] Cornelius, S. P., Kath, W. L., and Motter, A. E. *Nature Communications* **4** June (2013).

[116] Cowan, N. J., Chastain, E. J., Vilhena, D. A., Freudenberg, J. S., and Bergstrom, C. T. *PLoS ONE* **7**(6), e38398 June (2012).

[117] Delpini, D., Battiston, S., Riccaboni, M., Gabbi, G., Pammolli, F., and Caldarelli, G. *Scientific Reports* **3** April (2013).

[118] Galbiati, M., Delpini, D., and Battiston, S. *Nature Physics* **9**(3), 126–128 (2013).

[119] Jia, T. and Barabási, A.-L. *Scientific Reports* **3** August (2013).

[120] Jia, T., Liu, Y.-Y., Csóka, E., Pósfai, M., Slotine, J.-J., and Barabási, A.-L. *Nature Communications* **4** June (2013).

[121] Li, Z., Ren, W., Liu, X., and Fu, M. *International Journal of Robust and Nonlinear Control* **23**(5), 534–547 March (2013).

[122] Nicosia, V., Criado, R., Romance, M., Russo, G., and Latora, V. *Scientific Reports* **2** January (2012).

[123] Onnela, J.-P. J. *Science* **343**(6177), 1325–1326 March (2014).

[124] Pósfai, M., Liu, Y.-Y., Slotine, J.-J., and Barabási, A.-L. *Scientific Reports* **3** January (2013).

[125] Wang, W.-X., Ni, X., Lai, Y.-C., and Grebogi, C. *Physical Review E* **85**(2) February (2012).

[126] Yuan, Z., Zhao, C., Di, Z., Wang, W.-X., and Lai, Y.-C. *Nature Communications* **4** September (2013).

[127] Powell, M. and Ansic, D. *Journal of economic psychology* **18**(6), 605–628 (1997).

[128] Calvó-Armengol, A., Patacchini, E., and Zenou, Y. *Review of Economic Studies* **76**(4), 1239–1267 October (2009).

[129] Aral, S. and Walker, D. *Science* **337**(6092), 337–341 June (2012).

[130] Lomi, A., Snijders, T. A., Steglich, C. E., and Torló, V. J. *Social Science Research* **40**(6), 1506–1520 November (2011).

[131] Denrell, J. *Science* **321**(5885), 47–48 July (2008).

[132] Forgas, J. P. and Williams, K. D. *Social influence: Direct and indirect processes*, volume 3. Psychology Press, (2001).

[133] Rendell, L., Boyd, R., Cownden, D., Enquist, M., Eriksson, K., Feldman, M. W., Fogarty, L., Ghirlanda, S., Lillicrap, T., and Laland, K. N. *Science* **328**(5975), 208–213 April (2010).

[134] Couzin, I. D., Ioannou, C. C., Demirel, G., Gross, T., Torney, C. J., Hartnett, A., Conradt, L., Levin, S. A., and Leonard, N. E. *Science* **334**(6062), 1578–1580 December (2011).

[135] Estrada, E. and Vargas-Estrada, E. *Scientific Reports* **3** October (2013).

[136] Rafiee, M. and Bayen, A. M. In *Decision and Control (CDC), 2010 49th IEEE Conference on*, 3877–3883. IEEE, (2010).

[137] Kar, S. and Moura, J. M. In *Signals, Systems and Computers, 2006. ACSSC'06. Fortieth Asilomar Conference on*, 276–280. IEEE, (2006).

[138] Jafarizadeh, S. and Jamalipour, A. *arXiv preprint arXiv:1201.4480* (2012).

[139] Tran, T. M. D. and Kibangou, A. Y. In *Control Conference (ECC), 2013 European*, 227–232. IEEE, (2013).

[140] King, A. J. *Behavioural Processes* **84**(3), 671–674 July (2010).

[141] Allen, J. P., Porter, M. R., and McFarland, F. C. *Development and Psychopathology* **18**(1), 155 (2006).

[142] King, A. J. and Cowlishaw, G. *Communicative & integrative biology* **2**(2), 147–150 (2009).

[143] Couzin, I. D., Krause, J., Franks, N. R., and Levin, S. A. *Nature* **433**(7025), 513–516 (2005).

[144] Eguíluz, V. M., Zimmermann, M. G., Cela-Conde, C. J., and San Miguel, M. *American journal of sociology* **110**(4), 977–1008 (2005).

[145] Ioannou, C. C., Singh, M., and Couzin, I. D. *The American Naturalist* **186**(2), 284–293 (2015).

[146] Kao, A. B., Miller, N., Torney, C., Hartnett, A., and Couzin, I. D. (2014).

[147] Rosenthal, S. B., Twomey, C. R., Hartnett, A. T., Wu, H. S., and Couzin, I. D. *Proceedings of the National Academy of Sciences* **112**(15), 4690–4695 (2015).

[148] Valente, T. W. *Science* **337**(6090), 49–53 July (2012).

[149] Merkle, J. A., Sigaud, M., and Fortin, D. *Ecology letters* (2015).

[150] Chang, X., Tan, R., Xing, G., Yuan, Z., Lu, C., Chen, Y., and Yang, Y. *Parallel and Distributed Systems, IEEE Transactions on* **22**(8), 1407–1414 (2011).

[151] Felzer, K. R. and Brodsky, E. E. *Nature* **441**(7094), 735–738 (2006).

[152] Packer, A. M. and Yuste, R. *The Journal of Neuroscience* **31**(37), 13260–13271 (2011).

[153] Pigou, A. C. *The Quarterly Journal of Economics* , 38–65 (1917).

[154] Lancichinetti, A., Fortunato, S., and Radicchi, F. *Physical Review E* **78**(4), 046110 (2008).

[155] Rogers, E. M. *Diffusion of innovations*. Free Press, New York, (1983).

[156] Valente, T. W. *Social networks* **18**(1), 69–89 (1996).

[157] Valente, T. W. *Network models of the diffusion of innovations*, volume 2. Hampton Press Cresskill, NJ, (1995).

[158] Goldenberg, J., Han, S., Lehmann, D. R., and Hong, J. W. *Journal of Marketing* **73**(2), 1–13 (2009).

[159] De Nooy, W., Mrvar, A., and Batagelj, V. *Exploratory social network analysis with Pajek*, volume 27. Cambridge University Press, (2004).

[160] Herzog, W., Stanfield, J., and Whiting, G. *Diffusion of innovations research report* (10) (1968).

[161] Valente, T. W. *http://www-hsc.usc.edu/ tvalente/enp/enp.doc* .

[162] Alon, N. and Milman, V. D. *Journal of Combinatorial Theory, Series B* **38**(1), 73–88 (1985).

[163] Comellas, F. and Gago, S. *Journal of Physics A: Mathematical and Theoretical* **40**(17), 4483 (2007).

[164] Singh, P., Sreenivasan, S., Szymanski, B. K., and Korniss, G. *Physical Review E* **85**(4), 046104 (2012).

[165] Tanaka, G., Morino, K., and Aihara, K. *Scientific Reports* **2** January (2012).

[166] Klemm, K., Serrano, M. Á., Eguíluz, V. M., and Miguel, M. S. *Scientific Reports* **2** February (2012).

[167] Estrada, E. and Vargas-Estrada, E. *Applied Mathematics and Computation* **218**(21), 10393–10405 (2012).

[168] Erdös, P. and Rényi, A. *Publ. Math. Debrecen* **6**, 290–297 (1959).

[169] Albert, R. and Barabási, A.-L. *Reviews of modern physics* **74**(1), 47 (2002).

[170] Hagberg, A., Schult, D., and PJ, S. In *Proceedings of the 7th Python in Science Conference (SciPy2008)*, 11–15, (2008).

[171] Barabási, A.-L. and Albert, R. *science* **286**(5439), 509–512 (1999).

[172] Estrada, E. *Physical Review E* **82**(6), 066102 (2010).

[173] Snijders, T. A. In *Dynamic social network modeling and analysis: Workshop summary and papers*, 146–161. National Academies Press, (2003).

[174] Stumpf, M. P. and Ingram, P. J. *EPL (Europhysics Letters)* **71**(1), 152 (2005).

[175] Faloutsos, M., Faloutsos, P., and Faloutsos, C. In *ACM SIGCOMM Computer Communication Review*, volume 29, 251–262. ACM, (1999).

[176] Dorogovtsev, S., Mendes, J., and Samukhin, A. *Nuclear Physics B* **653**(3), 307–338 (2003).

[177] Malarz, K., Karpinska, J., Kardas, A., and Kulakowski, K. *arXiv preprint cond-mat/0309255* (2003).

[178] Blondel, V. D., Guillaume, J.-L., Hendrickx, J. M., and Jungers, R. M. *Physical Review E* **76**(6), 066101 (2007).

[179] Wiener, H. *Journal of the American Chemical Society* **69**(1), 17–20 (1947).

[180] Balaban, A. T. *Chemical Physics Letters* **89**(5), 399–404 (1982).

[181] Freeman, L. C. *Social networks* **1**(3), 215–239 (1979).

[182] Gutman, I., Yeh, Y., Lee, S., and Luo, Y. *Indian Journal of Chemistry* **32**, 651–661 (1993).

[183] Strogatz, S. H. *Nature* **410**(6825), 268–276 (2001).

[184] Wasserman, S. and Faust, K. *Social network analysis: Methods and applications* , 1–27 (1994).

[185] Devillers, J. and Balaban, A. T. *Topological indices and related descriptors in QSAR and QSPAR*. CRC Press, (2000).

[186] Caporossi, G. and Hansen, P. *Discrete Mathematics* **212**(1), 29–44 (2000).

[187] Mladenović, N. and Hansen, P. *Computers & Operations Research* **24**(11), 1097–1100 (1997).

[188] Hansen, P. and Mladenović, N. *European journal of operational research* **130**(3), 449–467 (2001).

[189] Aouchiche, M. *Personal communications* (2014).

[190] Hohn, F. E. *Elementary matrix algebra*. Courier Dover Publications, (1973).