Universidad
Rey Juan Carlos

DOCTORAL THESIS

Analysis and control of spectral centralities in graphs and hypergraphs

Author:

Gonzalo Contreras Aso

Supervisors:

Regino Criado Herrero

Miguel Romance del Río

Doctoral Program in Sciences
International Doctoral School

2024

Contents

List of Figures

List of Tables

Preface

Agradecimientos

Esta tesis es el fruto de tres años de trabajo bajo la supervisión de mis directores de tesis, Regino Criado y Miguel Romance. Les agradezco a ambos profundamente su eterna amabilidad y paciencia a la hora de lidiar con un físico, siempre dispuestos a discutir cualquier idea por absurda que pareciera, así como su predisposición a ayudar en lo que fuera necesario, y su guía en el complicado, a ratos tedioso, mundo académico.

En términos académicos también he de agradecer a Stefano Boccaletti el haberme brindado la oportunidad de participar en proyectos interesantes, que aun siendo tangenciales a esta tesis, me han suscitado un gran interés. Lo mismo puedo decir de Gabriel Mindlin y Jorge Tredicce; los tres han reavivado mi interés en la física. Por otro lado, quiero reconocer en especial el apoyo brindado por Karin Alfaro-Bittner estos tres años: poca gente está tan dispuesta a zambullirse contigo en innumerables líneas de código para sacar adelante las simulaciones.

Me gustaría además dar las gracias al resto del Laboratorio de Computación Matemática en Redes Complejas y Aplicaciones (LaCoMaRCA), Dani, David y Eva, así como al resto de compañeros y compañeras del departamento, por crear un ambiente agradable y amistoso que es verdaderamente envidiable (¿en qué otro departamento se resuelven ODS un viernes por la tarde?).

To finish the academic acknowledgements, I would like to thank my supervisors during my visit to Naples, Mario di Bernardo and Piero de Lellis, as well as the researchers I met over there, specially Kirill Kovalenko, whose mathematical discussions were very inspiring.

Queda agradecer a mi familia, especialmente a mis padres y mi hermano, por su incondicional apoyo todos estos años, aunque sigan sin tener muy claro en qué consiste mi trabajo; y a mis amigos y amigas, que siempre están ahí cuando les necesito.

Finalmente, esta tesis ha sido posible gracias a la financiación del programa propio de doctorado de la Universidad Rey Juan Carlos, concretamente a través del contrato predoctoral PREDOC21-026-2164.

Abstract

Network science has witnessed a surge in popularity in the past two decades. This research interest has transitioned from exploring standard networks to more intricate mathematical abstractions such as multilayer networks and hypergraphs. Centrality measures remain a cornerstone of complex network theory, with an active community that proposes theoretical advancements and applies them to real-world scenarios.

This thesis explores the controllability of spectral centrality measures in complex networks. This type of centrality measures is particularly important because of its analytical foundations and reduced computational cost, so much so that it currently underlies most Internet search engines. We analyze various techniques for manipulating these measures in graphs and multilayer networks, classifying the different centrality control paradigms based on the type of control exerted (structural or parametric) and the amount of control attained.

We then propose novel extensions of spectral centrality measures to non-uniform and directed/heterogeneous hypergraphs, overcoming the limitations of existing methods while retaining their mathematical consistency. These extensions leverage the Perron-Frobenius theory for tensors, thereby providing analytical guarantees of existence and uniqueness. Finally, we explore the controllability of the aforementioned generalization of spectral centralities under weight adjustments.

Although primarily theoretical, this work lays the groundwork for future research on controlling centrality measures and paves the way for investigating potential applications in real-world complex systems.

List of publications

This thesis is based on the following publications:

- **Gonzalo Contreras-Aso**, Regino Criado, Miguel Romance. "Can the PageRank centrality be manipulated to obtain any desired ranking?" In: *Chaos: An Interdisciplinary Journal of Nonlinear Science, Volume 33, 083152 (2023).*

  DOI: https://doi.org/10.1063/5.0156226

- **Gonzalo Contreras-Aso**, Cristian Pérez-Corral, Miguel Romance. "Uplifting edges in higher order networks: spectral centralities for non-uniform hypergraphs." Under review in: *SIAM Journal of Mathematics of Data Science*, preprint available in *arXiv:2310.20335 (2023).*

  DOI: https://doi.org/10.48550/arXiv.2310.20335

- **Gonzalo Contreras-Aso**, Regino Criado, Miguel Romance . "Beyond directed hypergraphs: heterogeneous hypergraphs and spectral centralities." In: *Journal of Complex Networks, Volume 12, Issue 4, cnae037 (2024).*

  DOI: https://doi.org/10.1093/comnet/cnae037

The author has further contributed to the following publications:

- David Aleja, **Gonzalo Contreras-Aso**, Karin Alfaro-Bittner, Eva Primo, Regino Criado, Miguel Romance, Stefano Boccaletti. "A compartmental model for cyber-epidemics." In: *Chaos, Solitons & Fractals, Volume 161, 112310 (2022).*

  DOI: https://doi.org/10.1016/j.chaos.2022.112310

- Pitambar Khanra, Subrata Ghosh, David Aleja, Karin Alfaro-Bittner, **Gonzalo Contreras-Aso**, Regino Criado, Miguel Romance, Stefano Boccaletti, Pinaki Pal, Chittaranjan Hens. "Endowing networks with desired symmetries and modular behavior." In: *Physical Review E 108, 054309 (2023).*

  DOI: https://doi.org/10.1103/PhysRevE.108.054309

- **Gonzalo Contreras-Aso**, Regino Criado, Guillermo Vera de Salas, Jinling Yang. "Detecting communities in higher-order networks by using their derivative graphs." In: *Chaos, Solitons & Fractals, Volume 177, 114200 (2023).*

  DOI: https://doi.org/10.1016/j.chaos.2023.114200

- Atiyeh Bayani, Fahimeh Nazarimehr, Sajad Jafari, Kirill Kovalenko, **Gonzalo Contreras-Aso**, Karin Alfaro-Bittner, Ruben J. Sánchez-García, Stefano Boccaletti. "The transition to synchronization of networked systems." In: *Nature Communications 15, 4955 (2024).*

  DOI: https://doi.org/10.1038/s41467-024-48203-6

Chapter 1
Introduction

> Mathematicians do not study objects, but the relations between objects; to them it is a matter of indifference if these objects are replaced by others, provided that the relations do not change.
>
> Henri Poincaré

By the end of the 20th century, science had conquered the realms of the big and the small: General Relativity was put forward by Einstein in 1915 and remains uncontested even today, and the Standard Model of particles has been around since the 1970s, only confirming already expected results rather than providing new ones. However, most phenomena at intermediate scales have complicated behavior, which is yet to be fully understood by analytic methods. From genes to the functioning of the brain, from flocks of birds to social interactions, there is a plethora of systems which feature individual constituents that seem to work in unison to perform extraordinary tasks. In most cases, the collective behavior and properties of the system are decoupled from the rules that each individual follows. A good example of this can be a fluid: each molecule would obey Newton's equations of motion, but collectively one finds new properties, such as viscosity or vorticity.

The study of complex systems, that is, systems where there is an emergent or collective behavior, has been named "Complexity Science", and it is a multidisciplinary endeavor involving mathematicians, physicists, biologists, computer scientists, sociologists, and so on. This research avenue has received considerable attention in recent years, and even decades, because of its wide range of applications across different fields. Models of brain synchronization have been used to study the onset of epileptic seizures [1], centrality measures have been used to provide relevant results in web searches [2], circadian rhythms have been discussed as systems of coupled oscillators [3], epidemic spreading models have been used to quantify and forecast

epidemic outbreaks [4], etc. Simultaneously, the democratization of computing power (the widespread availability of personal computers and programs to run within them) has spearheaded many of these findings because of the possibility of performing numerical explorations of these systems prior to understanding them analytically.

The fact that this is a multidisciplinary field also means that there is a vast array of tools and techniques available to study complex systems; for instance, researchers with a background in statistical physics are accustomed to dealing with lattice models, such as the quintessential Ising model, which has proven useful for understanding many systems [5] outside the original scope. Each system will benefit from a description in terms of a particular set of tools, which may shed more light on it than if we had used different ones. In this regard, the field of complex networks has been extremely fruitful and successful for the description of systems where there is an underlying, non-trivial connectivity between the interacting units. Some examples of these are very natural: the brain is a network of interconnections between neurons, the Internet is, as the name suggests, a network of interconnected webpages, and a collection of human beings interact between themselves, creating ties. But there are plenty of subtler examples, if one knows where to look: the brain has an underlying "functional" connectivity network between different regions with similar activations [6], flocks of birds can be thought of as networks where the connections determine which neighbors does each bird see [7], and even time series (e.g. the stock market prices) can be described as networks under certain transformations [8].

It is within the realm of complex network theory that this thesis is localized.

1.1 Brief timeline of the field of complex networks

The field of complex networks is fairly recent; it has been around 20 years now, which is not much. However, it has fed from much older fields. Primarily, it began as an extension of graph theory, thus drawing from it many results and techniques.

Graph theory itself had contributions from many areas of mathematics, such as topology, combinatorics and discrete mathematics. These are also invaluable in the field of complex networks. In fact, on many occasions, the results needed in this field already exist in those areas, and obtaining them is more a translational task than an analytic one. For instance, a property interesting to higher-order networks may already have been studied in combinatorics, but in a "different mathematical language", as each area has its particularities.

Other than graph theory, there is a huge contribution to the field coming from physics, especially statistical physics. Traditional concepts in physics such as entropy or free energy [9], or even dynamics [10] are brought to the realm of complex networks, where they have proven to be very useful and interesting to study.

There have been essentially three, decade-separated ideas which have boosted the interest in complex networks.

1. The first one was the realization, at the end of the 20th century, that not all vertices are equally relevant in a graph [11, 12]; in fact most real networks

follow a vertex degree power law, with few having high degree while most of them having one or two connections. Soon plenty of research began to pile up, explaining a plethora of phenomena which can be described in terms of these networks [13].

2. The next great idea involved different types of edges, i.e. the realization that edges may have different nature depending on the relation itself. For instance, in a social network some relations can be friendships, while other relations can come from co-working, family, etc. This is just an example of a system whose structure is richer than that of a simple network, a realization that gives rise to the concept of multilayer networks [14, 15].

3. The most recent breakthrough was considering the possibility that interactions themselves need not be pairwise. The simplest example of this is a group of three friends, which is not well captured by just three pairwise edges, we need a triangle-like interaction joining the three of them at the same time. This has given rise to an enormous flux of work in relation to these "higher order networks" [16, 17], which is still ongoing.

Besides these ideas, the most important boost to the field of complex networks has come from the dramatic increase in computational power over the last decades. Algorithms previously thought unusable for its computational requirements have become the bread and butter of the field, and increasingly larger networks can be analyzed with the advent of supercomputing, computer clusters, multiprocessing paradigms, etc. In fact, even networks amounting for hundreds of gigabytes of data can be dealt with by personal computing machines.

1.2 The quantification of importance

One of the most consolidated areas of study within the field of complex networks is that of *centrality measures*. It is very appealing from a practical point of view: understanding *quantitatively* which are the key elements participating in the system under question and the role they play. These studies were actually started by sociologists [18, 19, 20] long before mathematicians or physicists took interest in it. Their motivation was as simple as understanding the social interactions among groups of people and distinguishing between leaders and followers within those groups. They were among the first scientists to recognize the underlying networked structure of social interactions and dynamics, and they developed quantitative techniques to solve these problems. The explosion of popularity and success in the field of complex networks brought many of these studies into light, acknowledging their role in defining centrality measures that are still being actively researched.

This all would probably have gone unnoticed if it were not for Sergei Brin and Larry Page publishing their seminal paper in 1998 [21], where they developed an algorithm of what later would be recognized as a network centrality measure, based partly on linear algebra and partly on the theory of stochastic processes, which they

would then use in their search engine, Google. The massive success of Google as a search tool fueled the interest in networks and centrality measures (especially *spectral* ones), something that has been present ever since [22].

Currently, there is a plethora of centrality measures available, depending on the type of information they aggregate (for instance, there are local measures that take into account the immediate surroundings of the individuals within the system, or global measures, which take into account all paths or even regions of the network to assess importance), as well as the type of techniques they use to compute the actual centrality scores [23]. Each is suited to describing different phenomena, and the assessment of the sensibility of a measure for the system under question usually requires a deep understanding of the system. If we leave aside applications, theoreticians working on centrality measures are often concerned with devising new metrics suitable for different mathematical structures or satisfying different heuristics (i.e., criteria for the assessment of importance).

1.3 Structure and scope of this thesis

The purpose of this thesis is twofold. On the one hand, this thesis is devoted to a thorough, quantitative analysis of spectral centrality measures, in particular focusing on their controllability properties [24]. If we compute a centrality measure on the graph *as is*, we obtain a set of centrality scores, one per node, quantifying their relevance in the network based on the heuristics behind the measure. However, perhaps we want to impose some a priori scores on the nodes of the network, and for that, we can either exert some changes on the network or on the measure itself. But, as we will see, not all changes allow us to impose these outcomes. The extent to which they do so is what we refer to as *controllability*.

The landscape of possible changes one can think of is vast: it is not our intention to review every possible type of change, on every possible network and centrality measure. We have chosen a special kind of centrality measures, the so-called *spectral* centrality measures [25], which are among the most useful ones. We have also chosen certain changes that have not been studied previously, and where we can actually make analytical progress.

On the other hand, this thesis provides some completion in the recently developed area of spectral centrality measures in hypergraphs. As we will discuss at length, hypergraphs are the latest mathematical structure brought to the field of complex networks to generalize standard pairwise networks [16, 17], in the hope of more faithfully modeling many real systems. As with any generalization, they come with their own toolset and challenges. In particular, when it comes to spectral centrality measures, only the bare minimum was laid down in [26], with two major shortcomings: the constraint to uniform hypergraphs and the lack of a framework to study hypergraphs beyond undirected ones. In this thesis we present fixes to both issues, something which will enable us to bring back and discuss the original goal of controllability in this setting as well.

Regarding the structure of this thesis, we will begin by introducing in Chapter 2

some basic concepts and definitions from graph theory and complex network theory, as well as discussing in depth the subtleties of centrality measures, focusing on spectral ones. In Chapter 3 we establish the two main paradigms of centrality control, and we delve into each of them. In Chapter 4 we provide an overview of the state-of-the-art hypergraph spectral centralities and we then proceed to amend some of its limitations. We end in Chapter 5 with some conclusions and outlook.

1.4 Notation and conventions

In this section we will establish a consistent notation for the upcoming chapters, some of which will be reintroduced when discussed in throughout the main text. We start by specifying some recurrent symbols in Table 1.1.

Symbol	Meaning
G	Graph
$\mathcal{G}$	Multilayer network
H	Hypergraph
V	Node set
E	(hyper)edge set
N	Number of nodes
M	Number of layers (multilayer)/maximum order (hypergraph)
D	Degree matrix
L	Laplacian matrix
A	Adjacency matrix
P	Row-normalized adjacency matrix
I	Incidence matrix
H_G	Head matrix
T_G	Tail matrix
d_{ij}	Shortest distance from i to j
$k_i, k_i^{\text{in}}, k_i^{\text{out}}$	Degree, in-degree, out-degree
$\mathcal{T}$	Adjacency tensor
$(\cdot)^T, (\cdot)^t$	Transposition (matrix/vector), transposition (tensor)
$\mathcal{H}, \mathcal{Z}$	Type of tensor eigenvectors

Table 1.1: List of common graph and hypergraph-related symbols.

We denote vectors by boldface symbols/letters. We refer to the space of N by M real-valued matrices as $\mathbb{R}^{N\times M}$.

We will denote the canonical basis of $\mathbb{R}^N$ as $\{\mathbf{e}_1, ..., \mathbf{e}_N\}$, with canonical vectors is $\mathbf{e}_i = (0, \cdots, 1^{(i)}, \cdots, 0)^T$. The sum of all the canonical basis vectors will be frequently appearing, and thus is denoted as $\mathbf{e} = \mathbf{e}_1 + ... + \mathbf{e}_n = (1, ..., 1)^T$.

$\mathbb{I}_N \in \mathbb{R}^{N\times N}$ is the N-dimensional identity matrix, and $\text{diag}(\boldsymbol{v})$ is the diagonal matrix with the components of $\boldsymbol{v} \in \mathbb{R}^N$ as its diagonal elements. The notation $\boldsymbol{v} > 0$ means component-wise positive vectors, i.e. $v_i > 0\ \forall i = 1, ..., N$.

Lastly, $\mathcal{T} \in \mathbb{R}^{[m,N]}$ denotes a hypermatrix of size $\underbrace{N \times \cdots \times N}_{m}$.

We will also be frequently referencing vector norms. For every $p \in [1, \infty)$, the ℓ_p norm of $\boldsymbol{v}$ is defined as

$$||\boldsymbol{v}||_p = \left(\sum_{i=1}^{N} v_i^p \right)^{1/p}. \tag{1.1}$$

However, we will mostly be interested in the ℓ_1 and ℓ_2-norms, i.e.

$$||\boldsymbol{v}||_1 = \boldsymbol{v} \cdot \mathbf{e} = \sum_{i=1}^{n} v_i, \qquad ||\boldsymbol{v}||_2 = \left(\sum_{i=1}^{n} v_i^2 \right)^{1/2}. \tag{1.2}$$

Chapter 2
Graph theory and network science

Although the field of complex networks has drawn substantial amount of results from areas such as statistical physics, information theory, differential equations or even biology, there is no doubt that its foundations are solely graph-theoretic, as graph theory is the perfect framework to encompass the elements present in networks. It presents the necessary abstractions to quantify concepts such as interactions, paths, cycles, connectivity, etc. These and many more will be introduced in Section 2.1. We then shift our focus from pure mathematics to actual system modeling in Section 2.2. This will provide new structures and tools to supplement the graph-theoretical ones. Lastly, we will discuss at length a subset of such tools, which are referred to as centrality measures, in Section 2.3.

2.1 Graph theory basics

From the definition of a graph to the notion of distances or clusters within a network, there is a good amount of graph theory [27] involved in the understanding of networks and their features [28]. It is thus compulsory for us to start our journey here, to enable us to expand its field of view in the following sections.

2.1.1 Basic definitions

Everything in graph theory extends or is defined on top of the concept of a graph.

Definition 2.1 (Graph). *A graph $G = (V, E)$ consists of a non-empty, finite set of $N \in \mathbb{N}$ nodes $V = \{1, ..., N\}$ and a set of $L \in \mathbb{N}$ pairs of nodes $E \subseteq V \times V$, called edges or links.*

We will say that the graph is *undirected* if these pairs are symmetric, i.e $(i, j) \in E \Leftrightarrow (j, i) \in E$, otherwise we will say that the graph is *directed*. In the undirected case, all nodes connected to a given one are considered *neighbors*, or technically, adjacent to it. In the directed case one has to distinguish between node i's *parents* or *predecessors* (those pointing to i) and *descendants* or *successors* (those pointed to by i).

In many applications it is interesting to provide additional information to the graph, in particular it is quite common to study *weighted graphs*. A graph is weighted if its edges $(i, j) \in E$ have a number (weight) $w_{ij} \in \mathbb{R}$ associated to them. This number can represent a wide variety of things: for instance, it can represent the strength of the connection (more is stronger), or distance between the connected nodes (less is far apart). In terms of notation, weighted graphs are sometimes written as $G = (V, E, W)$ where $W : E \longrightarrow \mathbb{R}$ is a weighting function, although we will usually omit it for the sake of conciseness.

Something we must remark is the fact that, although graphs (directed or undirected) allow for the existence of self-loops, i.e. edges of the form (i, i), $i \in V$, we will often ignore this possibility. The same goes for multi-edges: for the most part we will only allow a single edge connecting any two nodes $i, j \in V$. Graphs without self-loops or multi-edges are referred to as *simple graphs* [29], which is the kind of graphs we will be dealing with most of the time.

Some distinguished graph types: Before moving on, we will present a collection of different types of graphs depending on their structure, which one often finds being discussed in the literature. In general, the graphs we will be working with throughout the rest of this dissertation are vastly bigger (in terms of the amount of nodes and edges they contain) and messier, in the sense that they will never be this simple.

However, the structures we will show now are relevant in a plethora of applications: for instance, when we discuss the PageRank algorithm, one way to think about it requires considering the complete graph of the original graph as a "teleportation layer" (see Section 2.3 for more details). Furthermore, the existence of this graphs as subgraphs in bigger networks is also relevant in many theorems (for instance, those related to graph planarity [29]) and certain applications (see Subsection 3.2.4).

a) *Complete graph*: graph containing all possible edges between nodes.

b) *Cycle graph*: connected graph where all nodes have only two edges, no more no less.

c) *Path graph*: connected graph where all nodes have at least one, at most two edges.

d) *Regular graph*: graph where all nodes have the same number of neighbors. (Note: cycle graphs and complete graphs are always regular, but the converse is not true)

e) *Star graph*: particular case of tree graphs, with N nodes, where $N - 1$ of them are only connected to the remaining one.

f) *Bipartite graph*: graph containing two disjoint sets of nodes which only have interconnections between the two but not intraconnections within.

g) *Tree graph*: graph with no cyclic subgraphs, or, in other words, a graph where any two nodes are connected by just one path. Its name comes from the fact that any of them can be represented as a branching tree, spanning down (or up) from a root, which can be any of its nodes.

See Figure 2.1 for examples of each one of these types of graphs.

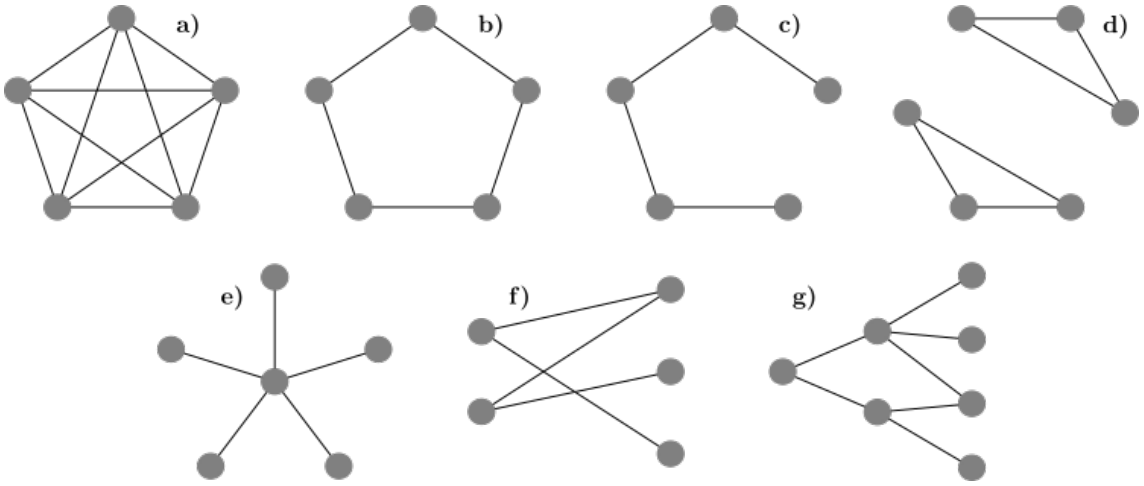


Figure 2.1: Examples of simple graphs described in the text: a) complete, b) cycle, c) path, d) 2-regular, e) star, f) bipartite, g) tree.

Some of these have similar directed counterparts (e.g. directed cycles graphs, directed trees, etc). Note also that although we have used the notion of "connectedness", we have yet to define it technically and beyond intuition.

2.1.2 Algebraic graph theory

As we discussed in the introduction, complex network theory draws from many different areas, and so does graph theory with other mathematical subjects whose methods have been proven useful in this context. Topological graph theory [30] and algebraic graph theory [31] are the main examples of this. We will now focus on the latter case, as it will be of utter importance throughout this dissertation.

Algebraic graph theory is concerned with the study of graphs using algebraic tools and methods. In particular, there are different ways to represent graphs using matrix theory, each with its own quirks and applications. Here we will give an overview of the most common of them.

The most common matricial representation of graphs is, by far, the so-called *adjacency matrix*.

Definition 2.2 (Adjacency matrix). *Let $G = (V, E)$, with $N = |V|$ the number of nodes. The adjacency matrix $A = (A_{ij}) \in \mathbb{R}^{N \times N}$ of G is the matrix with elements*

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

It should be clear from this definition that the adjacency matrix of an undirected graph is symmetric. This is generally not true for directed graphs.

If G is weighted, one can instead consider the *weight matrix* or *weighted adjacency matrix*, which is the same except having w_{ij} instead of 1 if the edge (i, j) is present in the graph. In an abuse of notation, we will use the term *adjacency matrix* even when we consider the weighted version, for the sake of conciseness.

A closely related matricial object is the incidence matrix of a graph. We will have to make a distinction, depending on the directedness of the graph.

Definition 2.3 (Undirected incidence matrix). *Let $G = (V, E)$ be an undirected graph, with $N = |V|$ the number of nodes and $L = |E|$ the number of edges. The incidence matrix $I = (I_{j\ell}) \in \mathbb{R}^{N\times L}$ of G is the matrix with elements*

$$I_{j\ell} = \begin{cases} 1 & \text{if} \quad \ell = (j, k) \in E \quad \text{for some} \quad k \in V, \\ 0 & \text{otherwise.} \end{cases} \tag{2.2}$$

Definition 2.4 (Directed incidence matrix). *Let $G = (V, E)$ be a directed graph, with $N = |V|$ the number of nodes and $L = |E|$ the number of edges. The incidence matrix $I = (i_{j\ell}) \in \mathbb{R}^{N\times L}$ of G is the matrix with elements*

$$I_{j\ell} = \begin{cases} -1 & \text{if} \quad \ell = (j, k) \in E \quad \text{for some} \quad k \in V, \\ 1 & \text{if} \quad \ell = (k, j) \in E \quad \text{for some} \quad k \in V, \\ 0 & \text{otherwise.} \end{cases} \tag{2.3}$$

Something that may come as a surprise is the fact that adjacency and incidence matrices are actually not related to linear algebra in the usual way. By this we mean that they are not to be thought of as linear operators on vector spaces. However, we can still use all tools in the linear algebra toolkit to operate with them or to even extract useful information from them (as we will see pretty clearly when we discuss spectral centralities in Subsection 2.3.3).

There are still a couple of interesting matrices to consider, but in order to do so first we should discuss what graph degrees are.

2.1.2.1 Node degrees

Having nodes on the one hand, and edges on the other, the first thing that comes to mind is, how many are "assigned" to each other? For instance, how many nodes are assigned to an undirected edge? This question is trivially two[1], no matter which edge we consider.

But when it comes to how many edges are "assigned" to a node, the answer is everything but obvious; clearly this depends on which node we discuss. This

[1] Or isn't it? As will be discussed in Subsection 2.2.4 and Chapter 4, the removal of this constraint has far-reaching consequences.

quantity is the degree of a node, usually denoted as k_i or $\deg(i)$. And we can use the adjacency matrix as a way to compute it, as it precisely contains the information of the adjacent nodes of any given one,

$$k_i = \sum_{j \in V} a_{ij} = \sum_{j \in V} a_{ji}. \tag{2.4}$$

Notice the last equality: that is true for undirected graphs, as their adjacency matrix is symmetric. However, if we discuss directed graphs, things change. In those cases we have to distinguish between the in-degree of a node k_i^{in} and the out-degree of a node k_i^{out}, as the number of incoming or outgoing links from node i, respectively. They are computed as

$$k_i^{\text{in}} = \sum_{j \in V} a_{ij}, \qquad k_i^{\text{out}} = \sum_{j \in V} a_{ji}. \tag{2.5}$$

It is possible to define the "total" degree of a node in a directed graph as $k_i = k_i^{in} + k_i^{out}$. Note also that in weighted graphs there are trivial generalizations of all these quantities (which are sometimes referred to as "weighted degree" or "strength" s_i [13]).

Node degrees can be aggregated in the *degree matrix*.

Definition 2.5 (Degree matrix). *Let $G = (V, E)$ be an undirected graph, with $N = |V|$. The degree matrix $D = (D_{ij}) \in \mathbb{R}^{N \times N}$ of G is the diagonal matrix containing the degree of each node in its corresponding entry in the diagonal,*

$$D_{ij} = \begin{cases} k_i & i = j, \\ 0 & i \neq j. \end{cases} \tag{2.6}$$

This matrix enables us to define another really important matrix, the so-called *graph Laplacian* or *discrete Laplacian*.

Definition 2.6 (Graph Laplacian). *Let $G = (V, E)$ be an undirected graph, with $N = |V|$. Let D be the degree matrix and A the adjacency matrix. The graph Laplacian $L = (L_{ij}) = D - A \in \mathbb{R}^{N \times N}$ of G is the matrix whose entries are*

$$L_{ij} = D_{ij} - A_{ij} = \begin{cases} k_i & i = j, \\ -1 & \text{if} \quad (i, j) \in E, \\ 0 & \text{otherwise.} \end{cases} \tag{2.7}$$

The graph Laplacian is a matricial representation of the graph which is extremely important in dynamical phenomena [10], playing a mayor role in, for instance, the synchronization of coupled oscillators.

So far we have introduced several algebraic objects (adjacency, incidence, degree and Laplacian matrices) and we have seen that they codify properties of the underlying

graph, such as the degrees, immediate neighbors, etc, but there are many other properties or results about graphs which can be extracted using algebraic techniques, some of which we will be discussing in the following sections, such as the connectivity of a graph (from the reducibility of its adjacency matrix, see Appendix A) or the amount of paths between nodes (from powers of the adjacency matrix). There are plenty of other algebraic properties which we will not be discussing, but which are of relevance in different areas, such as the relation between spectrum of the Laplacian matrix and the connectivity and synchronizability of the network [10].

2.1.3 Metric properties

So far we have discussed purely local properties of graphs: incidence and adjacency matrices are concerned with the immediacies of nodes and edges, with node degrees counting them. We will now explore other objects which arise when moving away from nodes.

We begin with some basic definitions of walks, trails, paths, and cycles [13, 32].

Definition 2.7 (Walk, trail, path, cycle). *Let $G = (V, E)$ be a graph, let $m \in \mathbb{N}$.*

- *A walk of length m from node $i \in V$ to node $j \in V$ is defined as an ordered collection of nodes $\{i_0 = i, i_1, i_2, ..., i_{m-1}, i_m = j\}$ such that $i_k \in V$ and $(i_{k-1}, i_k) \in E$ for every $k \in (1, m)$.*
- *A trail is a walk without repeating edges, i.e. $(i_{k-1}, i_k) \neq (i_{k'-1}, i_{k'})$ for $k \neq k'$.*
- *A path is a walk without repeating nodes, i.e. $i_k \neq i_{k'}$ for $k \neq k'$.*
- *A cycle is a closed trail, i.e. a trail with $i_0 = i_m$.*

These notions coincide with their expected, intuitive meaning, just with added rigor. We should remark that these definitions apply to both directed and undirected graphs, the only difference is that paths, walks, etc, can in general only be traversed in one way in directed graphs. These definitions also apply irregardless of the weighted nature of the network, although we will, as should be usual by now, discuss the unweighted versions for simplicity.

A very important notion which arises naturally from the previous definitions is that of *shortest paths*.

Definition 2.8 (Shortest path). *Let $G = (V, E)$ be a graph. A shortest path from node $i \in V$ to node $j \in V$ is a walk joining them with minimal length, therefore it is necessarily a path.*

Note we refer to shortest paths between nodes as "a" shortest path instead of "the" shortest path. This is due to the fact that they need not be unique, there may be several walks with the same minimal length between them. On the contrary, the shortest path length is actually unique.

A very important matrix which can be built from shortest paths is the *metric*, or *distance matrix*.

Definition 2.9 (Distance matrix). *Let $G = (V, E)$ be a graph, with $N = |V|$. The distance matrix $d = (d_{ij}) \in \mathbb{N}^{N\times N}$ of G is the matrix whose $d_{ij} \in \mathbb{N}$ element is the length of the shortest path between nodes $i \in V$ and $j \in V$. If there is none, then it is defined to be infinite.*

This matrix allows us to define further properties of the graph, such as its diameter ($\text{diam}(G) = \max_{i,j}(d_{ij})$) or the eccentricity of a node ($e(i) = \max_j(d_{ij})$). What's more it already glimpses at an important concept, which is that of connectivity, something we will explore below.

Lastly, note that the adjacency matrix A of the graph also encodes certain notions of distance. More explicitly, the entries of the m-th power of A count walks of fixed length m between nodes. To see this, consider the (i, j) entry of A^m:

$$(A^m)_{ij} = \sum_{k_1,k_2,\ldots,k_m=1}^{N} a_{ik_1} a_{k_1k_2} \ldots a_{k_mj}, \tag{2.8}$$

then each non-zero term in the sum corresponds to paths $\{i, k_1, k_2, \ldots, k_m, j\}$ existing within the network.

2.1.3.1 Connectivity

The above definitions enable us to establish different notions of *connectivity* in graphs. These will become utterly important in later chapters, especially as they play a major role in the Perron-Frobenius Theorem (see Appendix A), which ties it with algebraic graph theory [33]. Here we will have to make a distinction between directed and undirected graphs.

In undirected graphs, the definition of connectedness formalizes the intuition that a connected graph is a graph where all nodes are reachable by traversing edges.

Definition 2.10 (Connected graph). *Let $G = (V, E)$ be an undirected graph. G is connected if and only if there is a path from any node $i \in V$ to any other node $j \neq i \in V$.*

In directed graphs direction matters: node i might reach node j but perhaps not viceversa. Hence there are two different notions of connectedness in this case:

Definition 2.11 (Strongly connected graph). *Let $G = (V, E)$ be a directed graph. G is strongly connected if and only if there is a path from any node $i \in V$ to any other node $j \neq i \in V$.*

In other words, if all nodes are reachable from any node by traversing edges in their specific direction, then the graph is strongly connected. If we allow traversing

directed edges while ignoring their direction, we get the second notion of connectivity in directed graphs.

Definition 2.12 (Weakly connected graph). *Let $G = (V, E)$ be a directed graph. G is weakly connected if replacing all of its directed edges with undirected edges produces a connected graph.*

Related to these notions, a general measure of the connectivity of a network is the *average shortest path length*, sometimes also referred to as the *characteristic path length*.

Definition 2.13 (Average shortest path length). *Let $G = (V, E)$ be a graph, with $N = |V|$. Let d be its distance matrix. The average shortest path length is*

$$d_{\text{avg}} = \frac{1}{N(N-1)} \sum_{i,j=1}^{N} d_{ij}. \tag{2.9}$$

This quantity diverges for a not connected undirected graph or for a not strongly connected directed graph [9]. If it is finite, then the lower it is the "better connected" it is on average.

2.1.4 Further graph-theoretical notions

Most of network theory is built on top of what has already been discussed in the previous sections. However, there is plenty more graph theory that we left out (node/edge colorings, planarity, graph algorithms, etc), and there are a couple of items which are worth mentioning.

On the one hand, we will discuss the concept of *line graphs*, which will play a significant role later on. On the other hand, we will very briefly comment on some of the most classic graph-theoretical problems, Eulerian and Hamiltonian paths and walks, which concern pure mathematicians but are of less relevance to network theorists (even though we will use them).

2.1.4.1 Line graphs

For many applications it is interesting or even useful to shift our perspective, placing the focus of our study on the edges of the graph rather than on the nodes. For instance, consider the map of a city, where each street is an edge, and nodes represent intersections. If we want to study characteristics of the streets themselves, in many cases it is far more powerful considering them as nodes of some other related graph, whose features (such as their importance, as we will discuss in the next chapter) are easier to obtain.

One of the ways to formalize this idea is with the concept of the *line graph* of a graph [34].

Definition 2.14 (Line graph). *Let $G = (V, E)$ be an unweighted graph. Its line graph $L(G) = (V', E')$ is the graph whose nodeset is the edgeset of the original graph, i.e. $V' = E$, and its edge set consists of $((i, j), (j, k)) \in E'$ if $(i, j), (j, k) \in E$.*

Essentially, edges in the original graph G become the nodes of the line graph $L(G)$, which are connected in they share a node in G. Note that this definition is valid both for undirected as well as for directed graphs. In the latter case the node shared by two edges in G must be the start of one and the end of the other, for it to translate to an edge in $L(G)$.

This construction is useful in many proofs regarding properties of the original graph, as Whitney proved [35] that, with two exceptions, if two line graphs are isomorphic then their two original graphs are isomorphic as well (the converse is, however, not always true). Therefore, structural properties of the graph may pass on to the line graph, and it may be easier to compute them.

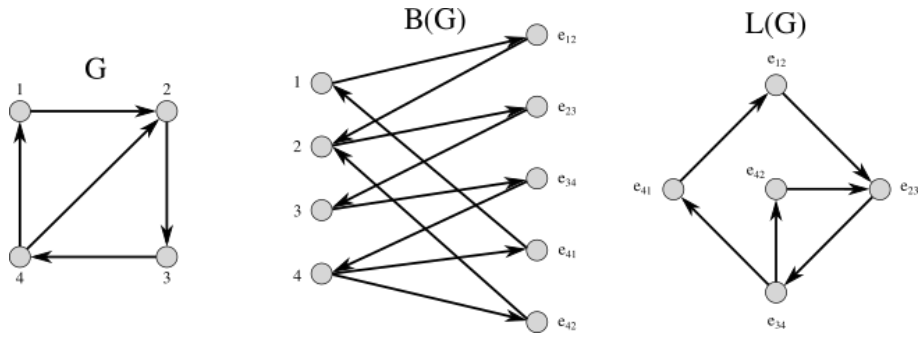


Figure 2.2: Example of a directed graph G along with its associated bipartite graph $B(G)$ and line graph $L(G)$.

From an algebraic point of view there is a relation between the adjacency matrices of a graph and that of its line graph, mediated by the bipartite graph $B(G) = (V^B, E^B)$ whose set of nodes is $V^B = N \bigcup E$ and elements of its edgeset are $(i, (i, j)), ((i, j), j) \in E^B$ if $i, j \in V$, $(i, j) \in E$. See Figure 2.2 for an example. The adjacency matrix of such bipartite graph is

$$A_{B(G)} = \left(\begin{array}{c|c} 0 & H_G \\ \hline T_G^t & 0 \end{array}\right) \in \mathbb{R}^{(N+L)\times(N+L)}, \tag{2.10}$$

where $H_G = (h_{ij}) \in \mathbb{R}^{N\times L}$ and $T_G = (t_{ij}) \in \mathbb{R}^{L\times N}$ are the incidence matrices of heads and tails[2], of G, respectively, defined by

$$h_{ij} = \begin{cases} 1 & \text{if} \quad \ell_j = (i, \cdot), \\ 0 & \text{otherwise,} \end{cases} \qquad t_{ij} = \begin{cases} 1 & \text{if} \quad \ell_j = (\cdot, i), \\ 0 & \text{otherwise.} \end{cases} \tag{2.11}$$

The relation between adjacency matrices is then given by [36]

$$(A_{B(G)})^2 = \left(\begin{array}{c|c} A_G & 0 \\ \hline 0 & A_{L(G)} \end{array}\right). \tag{2.12}$$

[2]This notation will appear again when we discuss directed hypergraphs.

Most of what we will be discussing in the next chapters will be based on spectral properties of adjacency matrices, which is why this relation is important. Moreover, it provides a way to generalize the concept of a line graph to that of a weighted graph: change the definition of H_G, T_G to include weights in such a way that the adjacency matrix of the weighted graph appears in $(A_{B(G)})^2$ and you get the adjacency matrix of the weighted line graph for free.

2.1.4.2 Eulerian and Hamiltonian paths

In 1736, Euler's resolution of the "seven bridges of Königsberg" problem [37] kick-started the field of graph theory, laying the foundations for everything that we've discussed throughout this chapter.

As a brief summary of the problem, the city of Königsberg (now Kaliningrad) had seven bridges over the Pregel river, connecting four parts of the city. Before Euler made an appearance, it was unclear whether one could design a walk that traversed each bridge once and only once. By abstracting the city and its bridges to a graph with 4 nodes (each part of the city) and 7 edges (the bridges), he came up with a negative solution: there is no walk on it passing through every bridge only once.

The following definitions are natural in general graphs.

Definition 2.15 (Eulerian path and graph). *Let $G = (V, E)$ be a graph. An Eulerian path is a walk on G visiting all edges once and only once. A graph which admits such a path is called an Eulerian graph.*

For our purposes the Eulerian property is not as important as the Hamiltonian property, which requires visiting all nodes once and only once.

Definition 2.16 (Hamiltonian path and graph). *Let $G = (V, E)$. A Hamiltonian path is a path on G visiting all nodes once and only once. A graph which admits such a path is called an Hamiltonian graph.*

As we will see in Chapter 3, this definition underlies the connectivity structure of the network, and in fact we will be able to use a modification of it to solve some structural problems which are of interest.

2.2 From graphs to networks

As discussed in the introduction, the field of complex networks (also known as network science) has its roots in the previously discussed mathematical formalism of graph theory, however around the early 2000's, it branched out of it due to, mainly, two reasons: firstly, graph theoreticians study more "formal" properties of graphs, such as node/edge colorings, connectivity, planarity or graph isomorphisms, whereas network theorists are interested in applications, where graphs are used to shed light on the real systems they represent.

Secondly, graph theory advances are constrained to the rigor of mathematics, however complex networks' advances sometimes rely on numerical or statistical computations, as well as heuristic arguments. Nevertheless, as we will see there are plenty of fully rigorous results, despite being aimed at specific applications.

2.2.1 Network phenomenology

As we have just discussed, there is a different aim in the area of network science, which places applications and understanding of real systems at the heart of it. This "network phenomenology" is far-reaching and very varied in both approach as well as subject of study, but we can basically distinguish two types of focuses.

Structural focus: In this first type, the network itself is under question. Information about the system the network is representing is expected to lie within the network, gathered through some procedure or another. For instance, centrality measures (understanding which are the most important elements of the network, see Section 2.3 for a fully-fledged discussion), structural balance [38] or community detection [39, 40] belong to this category. What's more, some measures already discussed in the previous sections (e.g. average shortest path length) already count as structural measures.

Dynamical focus: In this other type, the network is not the main object under study, but rather it is the ground on which some other phenomena take place, providing a non-trivial topology upon which the results can dramatically differ. For instance, synchronization of dynamical units [10, 41, 42], epidemics spreading [4, 43], percolation [44] or even game theory on networks [45] are prime examples of this second type.

It is important to note that these are not excluding categories: in fact, one of the "Holy Grails" of network science is understanding the interplay between the two (quantifying how much does the dynamics depend on the structure and determining how does the structure change based on the dynamics).

2.2.2 Classic network models

Due to the closeness to real systems, articles in the field of complex networks usually rely on networks constructed from real data, gathered from a variety of sources (web scraping, downloading and pre-processing files, actual experimentation of all sorts...). Nowadays there are even collections of free-to-use networks which one can use if it is interesting for the application in mind, such as [46, 47, 48].

Still, for many analytical arguments, ease of use and comparison, it is convenient to have baseline models which resemble real networks. We are now going to discuss three classic network models, those appearing in Figure 2.3, deserving their own name due to its remarkable properties or applications.

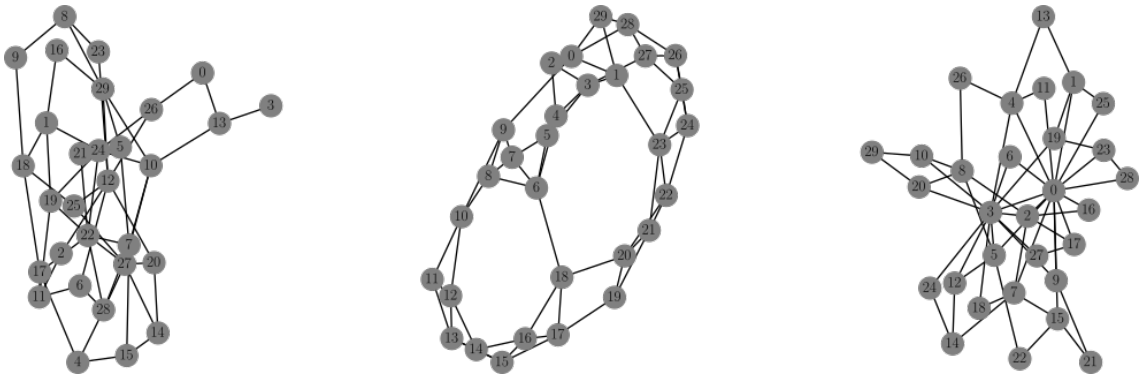


Figure 2.3: Three examples of networks: Erdős-Renyi $G_{30,0.12}$ (left), Watts-Strogatz $G_{30,4,0.15}$ (center) and Barabasi-Albert $G_{30,2}$ (right). Just from simple visual inspection one can see that the left one is mostly random, the center one resembles a ring with some added bridges, and the right one features a few nodes which have a noticeably higher degree than others.

Definition 2.17 (Erdős-Rényi random network). *Given $N \in \mathbb{N}$ and $p \in (0, 1)$, the Erdős-Rényi (ER) network $G_{N,p}$ is the undirected, unweighted graph generated with N nodes and with probability p for each edge (i, j) to exist.*

The Erdős-Renyi network model (sometimes abbreviated as *the ER model*) was proposed in [49], and is the most natural way to generate a random network. It is sometimes abbreviated as the ER model. Some important properties of it are the connectivity threshold $p_c = \log(N)/N$ (if $p < p_c$ the network will almost surely be disconnected, and vice-versa for $p > p_c$) [27].

This network model has been a cornerstone of network theory for decades, however its pure randomness is rarely seen in real networks. The reason for this is clear: real networks have guiding principles behind them (for instance, social relationships are built by humans through similarity and reasoning, the brain needs to enhance information flow, transport networks need to distribute traffic avoiding cascading failures, ...), not just randomness.

One of the first models for quasi-random networks with better connectivity was the one put forward by Duncan Watts and Steven Strogatz in [11].

Definition 2.18 (Watts-Strogatz small-world network). *Given $N \in \mathbb{N}$, $k \in \mathbb{N}$ and $p \in [0, 1]$, the Watts-Strogatz (WS) network $G_{N,k,p}$ is the undirected, unweighted graph generated with N nodes placed in a ring, then each connected to its k-nearest neighbors, and finally having each edge rewired at random with probability p.*

The Watts-Strogatz network model (sometimes abbreviated as *the WS model*) was the first successful attempt at generating probabilistically networks featuring the small-world property. This property is present in many real networks, in the form of having low average shortest path length compared to their size. If d_{avg} is the average shortest path length, the statement is $d_{avg} \sim \log(N)$ [11].

In this network model, the parameter p interpolates between a regular lattice ($p = 0$) and a random Erdős-Renyi network ($p = 1$). For intermediate values of p the small-world property arises, due to the appearance of "bridges" joining distant regions of the network, drastically reducing the average shortest path length.

While the Watts-Strogatz model was an accomplishment in the description of the small-world phenomenon, when contrasted with real networks it was clear that something was missing. In the Watts-Strogatz model nodes degrees are somewhat homogeneous, while in real networks there is a big difference between some nodes and the rest of them.

This difference can be stated more precisely, and it is often said that many real networks are "scale-free". What we mean by this is the fact that they have their degree distribution following a power-law $P(k) \sim k^{-\gamma}$, $\gamma > 0$ (and power laws are scale invariant, as $P(\alpha k) = \alpha^{-\gamma} P(k) \propto P(k)$). In other words, there are very few nodes with very high degree, while the majority of nodes have low degrees. The most prominent network model accounting for this is the one proposed by Albert-László Barabási and Réka Albert in [12].

Definition 2.19 (Barbási-Albert scale-free network). *Given $N \in \mathbb{N}$ and $1 \leq m < N$, $m \in \mathbb{N}$ the Barabási-Albert (BA) network $G_{N,m}$ is the undirected, unweighted graph generated adding N nodes one by one, and connecting each to m already present nodes, with the probability to choose node i to connect to being $p_i = k_i/(\sum_j k_j)$, with k_i the degree of node i.*

This growing network model features the *preferential attachment* mechanism: new nodes are more likely to be linked to already influential (in the sense of having many connections) nodes than to lesser connected ones. What's more, the model also features the small-world phenomenon, as the presence of hubs drastically reduces the length of shortest paths within networks.

This network model has had a massive success describing certain naturally occurring and human-made networks [50]. One of the reasons for this is the fact that these structures are very robust with respect to random failures (e.g. mutations in genetic code, disruptions in traffic/communication networks, ...), as opposed to ER networks, as the vast majority of nodes in a scale-free network are not hubs, and therefore their "disappearance" does not hinder the overall connectivity. On the other hand, scale-free networks are less robust when it comes to targeted attacks, as a malicious agent could choose to disrupt hubs, with drastic effects for the whole system [51, 43].

It should be noted that, despite the widespread use of these networks as realistic models for real-world systems, recently there have been some doubts about their ubiquity [52].

The three network models we have discussed here (ER, WS, BA) are the backbone of most of the network science literature, especially until the 2010's, when researchers realized that there are many systems whose modeling requires more elaborate structures beyond standard graphs.

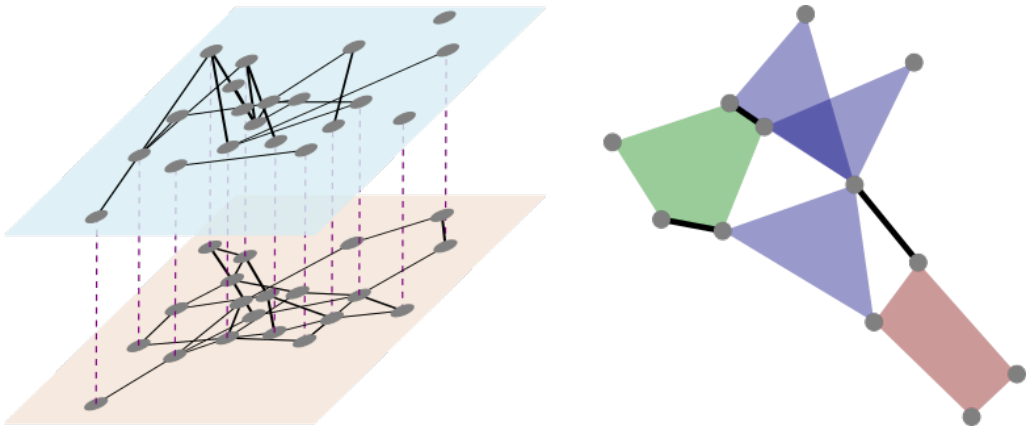


Figure 2.4: Examples of a multiplex network (left) with two layers and 30 nodes, and a hypergraph (right) with 12 nodes and interactions of orders 2 (black lines), 3 (light blue triangles), 4 (blue rhomboid) and 5 (dark blue pentagon).

2.2.3 Multilayer networks

In many real systems, interactions between elements are not of a single nature. For instance, networks of acquaintances feature connections such as friends, co-workers, family members... These can all be represented by edges but belong to different networks. In the early beginnings of network science these were all treated on equally, but soon researchers understood the need to accommodate for different types of interactions within a network, and thus multilayer networks appeared.

Here we will not spend much time discussing the ins and outs of multilayer networks, as they will make only brief appearances throughout. The interested reader is referred to [15, 14] and the references therein for a lengthy introduction and discussion on them. Nevertheless, let us explain how are some concepts from Section 2.1 generalized in this setting.

Definition 2.20 (Multilayer network). *A multilayer network is a pair* $\mathcal{M} = (\mathcal{G}, \mathcal{C})$ *where* $\mathcal{G} = \{G_\alpha;\ \alpha \in \{1, \dots, M\}\}$ *is a family of graphs* $G_\alpha = (V_\alpha, E_\alpha)$ *and*

$$\mathcal{C} = \{E_{\alpha\beta} \subseteq V_\alpha \times V_\beta;\ \alpha, \beta \in \{1, \dots, M\},\ \alpha \neq \beta\} \tag{2.13}$$

is the set of interconnections between nodes of different layers G_α *and* G_β *; with* $\alpha \neq \beta$.

The graphs G_α are often referred to as the layers of multilayer graph. The elements of each E_α (the edges within each layer), are called *intralayer* connections of $\mathcal{M}$ in contrast with the elements of each $E_{\alpha\beta}$, which are called interlayer connections.

For our purposes, this structure is too general; in later chapters we will focus on the study of centrality measures, which attempt to assign importance scores to nodes in networks, and in general multilayer networks the nodes in each layer may be completely different. A particular type of multilayer networks which is relevant in most applications is that of multiplex networks, defined below.

Definition 2.21 (Multiplex network). *A multiplex network is a multilayer network in which* $V_1 = V_2 = \cdots = V_M \equiv V$ *and* $E_{\alpha\beta} = \{(i,i); i \in V\}, \forall \alpha, \beta \in \{1, \ldots, M\}$, $\alpha \neq \beta$.

In other words, multiplex networks consist of a fixed set of nodes connected by different types of links. Typical examples of real multiplex networks can be social systems (individuals are related among themselves by friendship, coworking, or family ties, which would constitute separate layers), or brain connectomes (as synapses can be chemical or electrical, constituting them two layers with the same neurons in each one). One can see an example of such a structure in Figure 2.4.

Associated to each multiplex network there is the so-called *projection network* $\overline{\mathcal{M}}$.

Definition 2.22 (Projection network). *The projection network of a multiplex network is a graph* $\overline{\mathcal{M}} = (V, \overline{E})$ *where* $\overline{E} = \bigcup_{\alpha=1}^{M} E_\alpha$.

Many of the concepts defined in Section 2.1 for standard networks (graphs) can be generalized to the monoplex case in a fairly straightforward manner. For instance, the degree of a node in an undirected would now become a vector $\mathbf{k}_i = (k_i^{(1)}, k_i^{(2)}, \ldots, k_i^{(M)})^T \in \mathbb{R}^M$, where $k_i^{(\alpha)}$ is the degree of the node in layer α.

Of special interest for us are the adjacency matrices per layer $A^{(\alpha)} \in \mathbb{R}^{N \times N}$, which define the intralayer connectivity, as well as the adjacency matrix of the projection network, which is just

$$\overline{A} = \sum_{\alpha=1}^{M} A_\alpha. \tag{2.14}$$

2.2.4 Higher order networks

In recent years the community has realized the need to extend standard network-theoretical methods to a novel framework, that of higher order networks, which feature interactions beyond pairwise ones [16, 53, 17]. For instance, co-authorship networks where traditionally studied by creating links between authors who had collaborated together, but this construction neglects the fact that an article written by three authors is not the same as three articles written by the three pairwise combinations of them.

Systems with 2-interactions (e.g. pairwise), 3-interactions, etc can be described essentially using two distinct models: simplicial complexes and hypergraphs. The former is a more restrictive type, although in exchange providing a solid footing thanks to the mathematical area of simplicial homology, something which is out of the scope of this thesis (See [54, 55]). We will instead focus on the latter.

It turns out that there is an extension of graph theory to the hypergraph setting. We will now review the most basic results from algebraic hypergraph theory which will be interesting for our purposes, in preparation for Chapter 4.

We start with the definition of an undirected hypergraph.

Definition 2.23 (Undirected hypergraph). *A hypergraph $H = (V, E)$ consists of a non-empty set of N nodes $V = \{1, \dots, N\}$ and a set of hyperedges $e \in E$, $e \subseteq V$.*

An example hypergraph was shown in Figure 2.4. There are two details we should mention about this definition. Firstly, hyperedges $e \in E$ are usually sets of nodes, however in Section 4.2 we will relax this definition to multisets (which can include the same element more than once). This is analogous to the case of self-loops in graphs. Secondly, this definition leaves out any notion of directionality. Introducing this notion in hypergraphs is far from trivial, but at the same time it is something of significant relevance. This point will be discussed at length in Section 4.3.

As was the case with multilayer networks, most of the graph-theoretical concepts defined in Section 2.1 find some kind of generalization in the hypergraph context (hyperdegrees, connectivity, distances, etc). At the same time, there are new properties a hypergraph and its constituents can have. A particularly important one for us is uniformity.

Definition 2.24 (m-uniform hypergraph). *Let $H = (V, E)$ be a hypergraph and $m \in \mathbb{N}$. H is m-uniform if all of its hyperedges are of size m, i.e. $|e| = m$, $\forall e \in E$.*

The size of a hyperedge is sometimes referred to also as its order. Note that this subclass of hypergraphs is uncommon in real networks: if we consider the quintessential example of hypergraphs, which is the network of collaboration between scientists, the number of authors (nodes) in each hyperedge (paper) might not always the same. Note also that the case of 2-uniform hypergraphs coincides precisely with networks of pairwise interactions.

There are several analytical tools available for m-uniform hypergraphs which are generalizations of those discussed in Section 2.1. For instance, the notion of an incidence matrix (2.3) is still sensible in this new context. For us, the most relevnt structure to keep in mind is that of adjacency tensors[3].

Definition 2.25 (Adjacency tensor). *Let $H = (V, E)$ be a hypergraph, with $N = |V|$ the number of nodes. The adjacency tensor $\mathcal{T} = (T_{i_1 \dots i_m}) \in \mathbb{R}^{[m,N]}$ of H is the hypermatrix with elements*

$$T_{i_1 \dots i_m} = \begin{cases} 1 & \text{if } \{i_1, \dots, i_m\} \in E \\ 0 & \text{otherwise.} \end{cases}, \quad 1 \leq i_1, \dots, i_m \leq N, \tag{2.15}$$

where the generalization to weighted hypergraphs is straightforward.

Similarly to the graph case, in the context of undirected hypergraphs the tensors will always be symmetrical, meaning that $T_{i_1 \dots i_k} = T_{\sigma\{i_1, \dots, i_m\}}$, for all $\sigma \in \mathfrak{S}_m$, where

[3]The name "tensor" is usually reserved for mathematical objects invariant under coordinate transformations. In our case we are instead referring to multidimensional arrays (or hypermatrices), which we refer to as tensors for the sake of conciseness.

$\mathfrak{S}_m$ denotes the permutation group of m indices. The notion of directed hypergraphs will require a more careful treatment, something we will be discussing in Section 4.3.

As we discussed in standard graphs, algebraic properties of the adjacency matrix can be translated to topological/metrical properties of the graph. That same logic applies here, and in particular the strong connectivity of the hypergraph can be assessed from the irreducibility of its adjacency tensor. This is something we discuss in Appendix A, which will be of the utmost importance in Chapter 4.

2.3 Centrality measures

Out of all of the properties and features one would like to extract from a system, one of the most important is understanding the role each of its constituents plays. Is an individual very well connected to the rest? Is it isolated? Does it bridge groups which otherwise would not interact, perhaps because of being far apart?

When representing the system as a network, this problem and the answers to all of those questions are translated to the concept of *centrality* and its different variants. A node is central in a network if it is relevant in a specific way. In which way? That is something we must establish beforehand, by choosing a *centrality measure*.

Definition 2.26 (Centrality measure). *Let $G = (V, E)$ be a graph with $N = |V|$ nodes. A centrality measure is a mapping $f : V \to \mathbb{R}$ satisfying two properties:*

- *Centrality scores $f_i \equiv f(i)$ must be non-negative.*
- *The function f follows some heuristics, whose meaning is to be discussed.*

One usually requires the centrality scores to also be normalized, in other words

$$\tilde{f}_i = \frac{f_i}{\sum_{j \in V} f_j}. \tag{2.16}$$

.

The second property allows us to establish the kinds roles or importances the node should have in the network, which the measure should quantify. For instance, it is not always true that the node which is "best connected" is also the one closest on average to the whole network. This can be seen very clearly in Figure 2.5.

The study of centrality in complex networks is one of the most prominent ones, with some of the most impressive and remarkable results. Look no further than the PageRank algorithm, a way to rank webpages exploiting the networked nature of the Internet, which made Google the multi-million dollar company it is nowadays [2]. The applications of centrality measures go far beyond the Internet: certain centralities are the core of other phenomena, such as synchronization and coupled dynamics [56]; they have also been shown to be relevant in gene interaction networks [57], and some have even been seen to correlate with neuronal firing activity [58].

This Section is structured as follows: we will begin with an overview of some quintessential centrality measures in standard networks, namely, the degree, betweenness and closeness centralities. We will then move on to discuss spectral centralities,

which are the focal point of this thesis, hence we will spend some time understanding the main ones. We will finish with a brief summary of spectral centralities in multilayer networks, in preparation for Chapter 3. Hypergraph spectral centralities will be introduced and discussed in Chapter 4. Before moving on, let us stress the fact that most centrality measures are ill-defined if we consider negative weights, hence we will assume from here onwards unweighted or, at most, positively weighted networks.

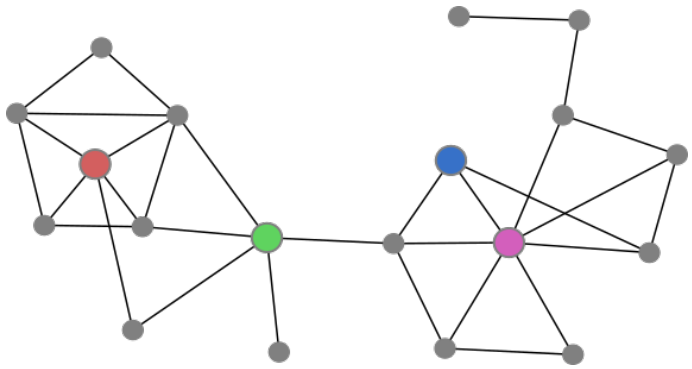


Figure 2.5: An example network, with the most central nodes (with respect to different centrality measures) highlighted, namely: eigenvector centrality (red), betweenness centrality (green), closeness centrality (blue) and degree centrality (pink).

2.3.1 Local centralities

Local centralities are, by far, the simplest (and perhaps uninteresting) measures of all. As their name suggests, they only depend on local properties of each part of the network, missing out on the topology or the structure as a whole.

As we will see in the later sections, features such as the quality of one node's connections or the closeness to every other node are relevant too, depending on the context. But it is nevertheless instructive to comment these basic measures before moving on.

2.3.1.1 Degree centrality

The most simple measure one can thing of is the *degree centrality*, which essentially measures how many connections does a node have. In an undirected graph it is computed as

$$f(i) = k_i, \qquad \bar{f}(i) = \frac{f(i)}{\sum_{j\in V} f(j)}. \tag{2.17}$$

In the context of directed graphs, we can actually split this centrality measure in two: the *out-degree centrality* $f^{out}(i)$ and the *in-degree centrality* $f^{in}(i)$, with straightforward extrapolations, whose node-wise sum is precisely $f(i)$ (up to normalization).

In the context of weighted graphs, one could instead consider the weighted version of the degree (also known as the *strength*, $f(i) = s_i$ and its directed counterparts).

An example of a situation where this measure shines is in the case of infectious diseases [4]. In these models, the most important element of the network in terms of contagion is that with highest degree, as that node or nodes will contribute more to the expansion of the disease.

The degree centrality, being the simplest, is also the most deceiving: in most cases the number of connections is not the only thing that matters. The main issue with this measure is that it is too local, it only counts nearest neighbors. But the topology of the network is utterly important.

2.3.2 Metric centralities

The word *centrality* itself reminds us of the idea of distance; the center of "something" is the middle point with respect to some distance-like measure. It shouldn't surprise then the fact that we can employ the tools developed in Subsection 2.1.3 to this particular problem. Here we briefly discuss the two most important centralities within this category.

2.3.2.1 Closeness centrality

A common idea about society is the fact that, if we pick a person at random from the whole (alive) humanity, on average the minimum people between anyone and that person is 6. That is on average: some people will require less than 6 [59] (probably presidents and important personalities), while the rest would need more than that. Clearly, the closer you are to the whole humanity, the more important you are, at least in this sense.

Computing the closeness of a node amounts to

$$f(i) = \frac{1}{\sum_{j\in V} d_{ij}}, \qquad \bar{f}(i) = (N-1)f(i). \tag{2.18}$$

There are many situations where this measure is utterly important; to give an example, if one is planning to set a distribution center for different shops in a city, you should decide where to place it by studying how close it is from each and every shop it will make deliveries to.

2.3.2.2 Betweenness centrality

There is an international saying which goes like "all roads lead to Rome". If that were true, Rome would be the most central node in the geography network, at least in the context of betweenness centrality.

This centrality measure characterizes how many least-distance paths go through a node.

$$f(i) = \sum_{j,k\neq i\in V} \frac{\sigma_{jk}(i)}{\sigma_{jk}}, \tag{2.19}$$

where σ_{jk} is the number of shortest paths from node j to node k, and $\sigma_{jk}(i)$ is the number of them which happen to go over node i.

In which situations is this measure important? Consider two islands joined by a bridge. If we model the roads inside and between the two islands as nodes, connected if the two roads cross each other, it is clear that a very central one will be the bridge joining both islands, even if it only has two edges.

This is nothing but a simple example of why this measure matters. In essence, whenever we are interested in the communication between different nodes of a network, this measure becomes one of the most relevant ones, as communications tend to follow geodesic (shortest distance) paths.

It is important to point out an serious flaw of both the closeness and the betweenness centralities: they both rely on computing shortest paths within the network. While this may be possible for reasonably small networks, for most ones with real applications it is a computationally intensive task. After all, finding all shortest paths has a computational complexity of $\mathcal{O}(N^3)$, or $\mathcal{O}(N^2 \log(N) + NL)$ if some other algorithm (e.g. Dijsktra) is used [60].

2.3.3 Spectral centralities

All of the previous centrality measures rely on either purely local features (the individual node degrees) or global metric ones (shortest paths). However, as we discussed in Subsection 2.1.2, there are several matrices representing the graph, the adjacency matrix being the most prominent one. This bears the natural question: could we leverage all the linear algebra machinery to extract information on important nodes based on this matrix? The answer is affirmative, and paves the way for a plethora of measures (known as *spectral centralities*) which span different types of networks, aims and even multimillion-dollar companies, as we'll see.

Here we will focus on the most relevant of these measures in standard networks, their mathematical foundations and features. Some of the measures discussed here will be essential for the coming chapters, namely the eigenvector centrality and PageRank. Others we will introduce for completeness' sake. For a more in-depth discussion and historical background, we refer the reader to [25].

To follow along most of what follows, it will be essential to have in mind the Perron-Frobenius Theorem [61, 62], and its graph-theoretical implications, both of which are discussed in Appendix A. Here is a quick recap of its immediate consequences: a graph is strongly connected if and only if it has an irreducible adjacency matrix, which in turn implies that there is a unique positive eigenvector associated to its spectral radius.

2.3.3.1 Eigenvector centrality

The problem with the degree centrality is that it is too short-sighted; it only looks at the number of nearest neighbors. Consider a simple idea: extending the degree centrality by incorporating in the sum (2.17) the centrality of the node's neighbours.

In other words, a node's importance being proportional to the importance of its nearest neighbors.

Formalizing this idea [63], we have

$$f(i) \propto \sum_{j \to i} f(j) \quad \Rightarrow \quad f(i) = \frac{1}{\lambda} \sum_{j \in V} a_{ji} f(j), \tag{2.20}$$

where we have introduced the proportionality constant λ, and the adjacency matrix $A = (a_{ij})$ to implement the nearest neighbors interaction.

Now, we can actually reformulate this problem in a more algebraic manner, using vectors $\mathbf{c}$ such that their components are $c_i = f(i)$. With this, we have

$$A^T \mathbf{c} = \lambda \mathbf{c}, \quad \mathbf{c} > 0. \tag{2.21}$$

This is the eigenvector equation of the transposed adjacency matrix, and thanks to Theorem A.2 we know that if we consider a strongly connected graph, it has a positive, unique (up to scaling) solution, therefore satisfying the axioms of a centrality measure (see Definition 2.26). Indeed, the Perron eigenvector, associated to the spectral radius $\rho(A)$, provides information on the importance of nodes in a network, based on their acquaintances.

It is natural to wonder what happens with weakly connected or even disconnected graphs. In both cases it is natural for non-local measures to break down (for instance, the lengths of shortest paths within these networks diverge), so we should not be surprised when it happens in this one. Still, we will see soon how PageRank manages to circumvent this issue.

2.3.3.2 Katz centrality

Consider a directed acyclic graph (DAG). In this kind of graphs there are some nodes which have zero in-degree, i.e. $k_i^{\text{in}} = 0$. If we were to naïvely apply the eigenvector centrality to this graph, their centrality score would be zero (they receive no contribution). This is in agreement with the Perron-Frobenius Theorem A.2, as this type of graph is, at most, weakly connected.

The Katz centrality [18] arises as simple modification to the standard eigenvector centrality which improves this issue. In order to do so, it provides each node with a base centrality score β, before adding the contribution from its neighbors. This modification in (2.21) reads

$$\mathbf{c} = \alpha A^T \mathbf{c} + \beta \mathbf{e}, \tag{2.22}$$

where $\mathbf{e} = (1, \ldots, 1)^T$.

Note that if $\beta = 0$, then we are back in the eigenvector centrality case (2.21) with $\lambda = 1/\alpha$. However, with $\beta > 0$ we no longer have an eigensystem, and thus α now becomes a parameter which we can tune. We can rewrite this equation as

$$\mathbf{c} = (\mathbb{I}_N - \alpha A^T)^{-1} \mathbf{e}, \tag{2.23}$$

provided we can invert that matrix. We cannot do so if $\det\left(\mathbb{I}_N - \alpha A^T\right) = 0$, i.e. if $1/\alpha$ is an eigenvalue of A^T. For $\alpha = 0$ there is no problem, but the centrality vector is uniform ($\mathbf{c} = \beta\mathbf{e}$). We can increase the value of α until we encounter the inverse of the maximum eigenvalue $\rho(A)$, at which point the previous matrix is singular. Therefore $\alpha \in [0, 1/\rho(A))$.

If we are within that range, there is another interpretation of the Katz centrality worth mentioning. We can express the previous equation as a Neumann series as follows

$$\mathbf{c} = \left(\sum_{n=0}^{\infty} \alpha^n (A^T)^n\right)\mathbf{e} = \left[\alpha^0 (A^T)^0 + \alpha^1 (A^T)^1 + \alpha^2 (A^T)^2 + ...\right]\mathbf{e}, \tag{2.24}$$

where convergence is guaranteed by the range of α.

This has a nice, alternative interpretation for the Katz centrality measure, namely that it acts as a modification to the standard eigenvector centrality considering more than just nearest-neighbors, taking into account the influence of each and every node, damped by powers of α based on their distance. For this reason, the parameter α is usually referred to as the "damping factor", and it is also a key ingredient in the next measure we are going to introduce.

2.3.3.3 PageRank

Larry Page and Sergei Brin, the creators of the Google Search Engine, designed in 1998 a way to rank Internet pages based on the networked nature of the web, with pages pointing to one another [21]. They did it in a way that, not only exploited powerful techniques from the field of Markov chains [64], but also circumvented the issue of the possible lack of strong connectivity, a drawback of the standard eigenvector centrality.

First of all, we need to define the *transition matrix* $P = (p_{ij})$ as the row-normalized[4] adjacency matrix A of a graph, i.e.

$$p_{ij} = \frac{a_{ij}}{\sum_{s\in\mathcal{N}} a_{is}} = \frac{a_{ij}}{k_i^{\text{out}}} \quad \text{if } k_i^{\text{out}} \neq 0, \text{ otherwise } p_{ij} = 0. \tag{2.25}$$

This transition matrix defines Markov chains (memory-less random walks) on the graph, as the entry p_{ij} provides the probability of, starting from node i, randomly choosing node j to visit next. More explicitly, it implements the following iteration

$$\mathbf{x}^{(k+1)} = P^T \mathbf{x}^{(k)}, \tag{2.26}$$

where $\mathbf{x} \in \mathbb{R}^N$ is a positive vector with unit ℓ_1-norm $||x||_1 = 1$, where x_i would represent the occupation of node i. The limit as $k \to \infty$ then yields the stationary distribution of random walkers on the network, i.e. the amount of times a random walker spends on each node. Denoting it as $\boldsymbol{\pi} \in \mathbb{R}^N$, we have

$$\boldsymbol{\pi}^T = \boldsymbol{\pi}^T P. \tag{2.27}$$

[4] It is sometimes defined instead as $D^\dagger A$ where D is the degree matrix of the graph (see Definition 2.5) and $\dagger$ represents the pseudoinverse, but either way the result is the same.

This is a reasonable measure of centrality for strongly connected graphs, also coming from a spectral analysis of a graph-based matrix, as the Perron-Frobenius Theorem A.2 also comes into rescue here. Furthermore, as P is row-normalized by definition, the eigenvalue associated to $\boldsymbol{\pi}$ is exactly 1, and that is the spectral radius.

However, the complete description of PageRank requires the introduction of two ingredients: a damping factor $\alpha \in (0, 1)$ as well as a "teleportation" or "personalization" vector $\boldsymbol{v} \in \mathbb{R}^N$ with $\boldsymbol{v} > 0$, $||\boldsymbol{v}||_1 = 1$. The former represents, in Markovian terms, the probability of the random walker of following the network edges (2.26), otherwise the walker would "teleport" to a node chosen at random from the latter (which is thought of as a probability distribution for the landing).

Piecing all these ingredients together we arrive at the PageRank equation for the stationary distribution $\boldsymbol{\pi} \in \mathbb{R}^N$, which defines the PageRank centrality.

$$\boldsymbol{\pi}^T = \boldsymbol{\pi}^T \mathbb{G}(\alpha, \boldsymbol{v}), \quad \text{where} \quad \mathbb{G}(\alpha, \boldsymbol{v}) = \alpha P + (1 - \alpha)\mathbf{e} \cdot \boldsymbol{v}^T, \tag{2.28}$$

where again $\mathbf{e} = (1, \ldots, 1)^T$. One can easily see that $\boldsymbol{\pi}$ is a left eigenvector of $\mathbb{G}(\alpha, \boldsymbol{v})$, the so-called "Google matrix". This matrix is, again, row-normalized, and it avoids the requirement of strong connectivity of the network as it is now a positive matrix due to the personalization term[5]. Hence we can again invoke the Perron-Frobenius Theorem A.2 to guarantee the existence and uniqueness of a positive solution to this equation, associated with the spectral radius $\rho(\mathbb{G}(\alpha, \boldsymbol{v})) = 1$ (due to normalization).

Now that we have seen the eigenvector centrality, the Katz centrality and PageRank, it is worth stepping back for a moment to have a global understanding of the relation between them, as shown in Table 2.1.

	No normalization	Row normalization
No damping	Eigenvector centrality (2.21)	Markovian spectral ranking (2.27)
Damping	Katz centrality (2.24)	PageRank without personalization (2.28)

Table 2.1: Relation between some of the most well-known spectral centrality measures (eigenvector, Katz and PageRank), as part of the same family, and only differentiated by either row normalization and the presence of damping [25].

PageRank exploded in popularity due to the success of the Google search engine, establishing itself as a paradigm of what the study of networks could yield. This popularity boosted plenty of research about it, both analyzing it (see for instance [65, 66, 2, 22, 67]) as well as extending it.

Some of these extensions of this measure include:

- **Biplex PageRank:** in [68] a novel version of the PageRank vector was put forward by establishing an analogy between the standard PageRank algorithm

[5]However, in directed networks, there can be "dangling nodes", which are nodes with zero out-degree. These are represented as a row with zeros in the transition matrix, and that can be problematic (for instance, for the irreducibility). This is often solved by substituting it with $P + \mathbf{d} \cdot \boldsymbol{u}^T$, where $\boldsymbol{d} = (d_i) \in \mathbb{R}^N$ is the vector indicating dangling nodes ($d_i = 1$ if i is a dangling node, otherwise $d_i = 0$) and $\boldsymbol{u} \in \mathbb{R}^N$, $||\boldsymbol{u}||_1$ is the dangling node distribution.

and a random surfer on a biplex network, which was shown to be useful in order to extend the notion of PageRank centralities to multiplex networks (even though here we will restrict ourselves to the standard network version).

In the biplex PageRank algorithm, the biplex network considered consists of two layers: one with the actual edge connections between the N nodes, the other containing a fully connected graph between them. We again have a damping factor $\beta \in (0,1)$ which specifies when to traverse the edge-connectivity layer or the teleportation one (the fully connected), the latter with chance of teleportation to each node given by yet another personalization vector $\boldsymbol{v} \in \mathbb{R}^N$ with $\boldsymbol{v} > 0$, $||\boldsymbol{v}||_1 = 1$.

That construction led to the definition of the Biplex PageRank $\boldsymbol{\pi}_{\text{BPR}}(\beta, \boldsymbol{v}) \in \mathbb{R}^N$ as the vector

$$\boldsymbol{\pi}_{\text{BPR}} = \boldsymbol{\pi}_u + \boldsymbol{\pi}_d > 0, \tag{2.29}$$

where $[\boldsymbol{\pi}_u^T \; \boldsymbol{\pi}_d^T] \in \mathbb{R}^{2N}$ is the unique normalized and positive eigenvector of

$$M_{\text{BPR}} = \begin{pmatrix} \beta P & (1-\beta)\mathbb{I}_N \\ \beta \mathbb{I}_N & (1-\beta)\mathbf{e} \cdot \boldsymbol{v}^T \end{pmatrix}. \tag{2.30}$$

Existence and uniqueness of the Biplex PageRank centrality are granted by the Perron-Frobenius Theorem A.2 [68].

- **Node-dependent restart PageRank:** in [69] a generalization of PageRank was put forward, where instead of considering a global damping factor they assume a node-dependent damping $\alpha_i \in (0,1)$, $i \in 1, ..., N$, which can be used to make further adjustments to the centrality output.

  The node-dependent restart PageRank vector $\boldsymbol{\pi}_{\text{NPR}}(\alpha_1, ..., \alpha_N, \boldsymbol{w})$ is defined as the positive, leading eigenvector $\boldsymbol{\pi}_{\text{NPR}} \in \mathbb{R}^N$

$$\boldsymbol{\pi}_{\text{NPR}}^T \left[\mathcal{A}P + (\mathbb{I}_N - \mathcal{A})\mathbf{e} \cdot \boldsymbol{w}^T \right] = \boldsymbol{\pi}_{\text{NPR}}^T, \quad \mathcal{A} = \text{diag}(\alpha_1, ..., \alpha_N) \tag{2.31}$$

  where $\mathcal{A} = \text{diag}(\alpha_1, ..., \alpha_N) \in \mathbb{R}^{N \times N}$ is the diagonal matrix containing each damping and $\boldsymbol{w} \in \mathbb{R}^N$ is again a personalization vector $\boldsymbol{w} > 0, ||\boldsymbol{w}||_1 = 1$.

  The introduction of this node-dependent damping α_i enables the discussion of different possibilities and cases of interest. The $\alpha_i = \alpha$, $\forall i$ limiting case corresponds to the usual PageRank. Another interesting case, quite realistic from an Internet surfer point of view, is that of the degree dependent restart: one chooses to follow links or jump elsewhere based on the amount of links available (and their variety, although that is harder to introduce). In the original paper [69] they entertain this possibility, introduced as

$$\mathcal{A} = \mathbb{I}_N - aD^\sigma, \tag{2.32}$$

  where D is the (out)-degree matrix and a, σ are tunable parameters.

2.3.3.4 Non-backtracking centralities

In its simplest form, the eigenvector centrality has a feature which blurs the actual importance of nodes, which is the presence of feedback (or backtracking, as displayed in the literature). By this it is meant that, as your centrality score is proportional to the centrality of your neighbours and your neighbours' centrality is proportional to yours, you will be given part of your centrality back, increasing your total score. This is not a flaw only affecting the eigenvector centrality: most spectral measures have it present, one way or another. However, spectral centralities have proven extraordinarily useful, therefore it may be interesting to compute them while removing this backtracking effect.

In [70] the authors identified this problem in the eigenvector centrality, and they showed how this artificially increases the centrality of the already most central nodes in a network. They provide a way to solve this problem by introducing the Hashimoto or non-backtracking matrix [71], defined as

$$B = (b_{i\to j,k\to l}) = \delta_{jk}(1-\delta_{il}). \tag{2.33}$$

This matrix is very reminiscent of the adjacency matrix of the line graph, which can actually be written in terms of its components as $A^{L_G} = (a^{L(G)}_{i\to j,k\to l}) = \delta_{jk}$. The difference lies then in the removal of backtracking in (2.33): in the line graph an edge is generated from two edges (i,j) and (k,l) if the end of the former is the start of the latter $j=k$, whereas in the graph whose adjacency matrix is the Hashimoto matrix the same applies, but only if the other two endpoints are different $k \neq l$. Hence, bidirectional edges are essentially suppressed, removing the feedback.

From (2.33) one can proceed as in [36], computing the Perron eigenvector of B (associated to edges), recovering the centralities of the nodes in the original graph by summing the centralities of edges starting from it:

$$c_i = \sum_{i\to k} c_{i\to k}, \tag{2.34}$$

where $c_{i\to k} > 0$ is the centrality associated to edge $i \to k$ obtained from the Hashimoto matrix.

This approach of [70] was, however, ill-defined in certain cases, as was shown in [72, 73]. The reason for this is because the removal of backtracking edges in the graph whose adjacency matrix is (2.33) may lead to a loss of (strong) connectivity. They then provide a modification to the original proposal, based not on edge removal but on edge damping, i.e.

$$B(\mu) = (b(\mu)_{i\to j,k\to l}) = \delta_{jk}(1-(1-\mu))\delta_{il}) \tag{2.35}$$

and then following the same steps as in the Hashimoto or [36] case. They prove [72] that this measure is well-defined for $\mu \in (0,1]$ (where in the limit $\mu \to 0$ it coincides with the original proposal, and if $\mu = 1$ it is the line graph version), and they also extend this procedure to the case of PageRank [73].

2.3.3.5 Spectral centralities in multilayer networks

With the surge of interest in multilayer networks, which we already introduced in Subsection 2.2.3, several extensions to the spectral centralities discussed in this section were put forward. It is not our purpose to review this whole landscape of measures, instead we are going to focus on one set of them, due to their simplicity and relevance in what's to come, those defined in [74].

Before beginning, let us note that the measures we will discuss apply to the multiplex case, i.e. multilayer networks where the nodeset is identical in all layers. And even if the nodesets are not identical, we can consider all nodes in all layers, just disconnected if not participating in one of those (for instance, a Twitter user without Facebook profile can be introduced in the Facebook layer as an isolated node). This is a reasonable restriction from a centrality point of view (if different nodes are present in different layers, how can we even account for this extra structural information?).

Consider then a multiplex network $\mathcal{G} = \{G_1, ..., G_M\}$, whose associated adjacency matrices per layer are A_m. We further assume a non-negative "cross-influence matrix" $Q = (q_{lm})$ such that q_{lm} quantifies how much does layer l influence layer m.

In this context, there are 4 different eigenvector-like centralities to consider [74].

- *Quasimonoplex*: the "independent-layer" and "uniform" eigenvector-like centralities can be defined, somewhat ignoring the multiplex context of the network.

  The first one ignores it altogether, as it just computes an eigenvector centrality per layer (provided they are all connected), yielding a vectorial centrality $\mathbf{c}_i \in \mathbb{R}^M$ to each node. The second one considers the eigenvector centrality of the projection network (see Definition 2.22).

  The first one requires all layers to be (strongly) connected, a rather restrictive requirement. The second one only requires the projection network to be strongly connected.

- *Heterogeneous*: the "local-heterogeneous" and "global-heterogeneous" eigenvector-like centralities take at face-value the interconnectedness of the layers, introducing it via the aforementioned cross-influence matrix Q.

  The local-heterogeneous centrality computes the Perron eigenvector of the adjacency matrices $A^\star_m$ of some weighted-projection networks (where the weighting is given by Q), yielding yet again a vectorial centrality $\mathbf{c}_i \in \mathbb{R}^M$ to each node.

$$A^\star_m = \sum_{l=1}^{M} q_{ml} A_l, \Rightarrow \quad (A^\star_m)^T \mathbf{c}^{(m)} = \lambda^\star \mathbf{c}^{(m)}. \tag{2.36}$$

  The global-heterogeneous centrality computes the Perron eigenvector of the matrix $A^\otimes$ given by the Khatri-Rao product [75] of the matrices Q and $(A_1|A_2|...|A_M)$.

$$A^\otimes = Q \otimes (A_1|A_2|...|A_M) \Rightarrow \quad (A^\otimes)^T \mathbf{c} = \lambda^\otimes \mathbf{c}. \tag{2.37}$$

Both of them have the same requirement as the uniform eigenvector-like centrality measure presented before, i.e. the projected graph $\overline{G}$ is required to be strongly connected.

Chapter 3
Controlling centrality measures on networks

A very interesting question to ask regarding the centralities in a network is whether such centralities are controllable[1] or not. By this we mean that, given a network $G = (V, E)$ with $N = |V|$ nodes, and an arbitrary centrality vector $\mathbf{c} \in \mathbb{R}^N$, can we modify "something" to allow for $\mathbf{c}$ to be its centrality vector?

The answer is, it depends. It turns out that we can consider very different versions of this problem based on which centrality measure we are considering, which changes are allowed, and how much control is exerted over the centrality outcome. In this chapter we explore these problems in some detail, establishing different theorems, bounds and counterexamples which furnish a catalog of possible centrality control schemes in standard networks and multilayer graphs.

It is worth mentioning that, even though we will always frame the discussion in the context of network centrality measures, the kind of problems we will be attempting to solve here are also of a deeply algebraic nature. For instance, in Section **??**, one of the problems we will investigate is the question of full control in directed graphs with weight changes. This problem can stated as

Problem 3.1 (Full Control (version 1)). *Given a strongly connected network G, can we modify its weights such that we achieve any possible centrality vector?*

This problem can be alternatively phrased as

Problem 3.2 (Full Control (version 2)). *Can the Perron eigenvector of an arbitrary non-negative, irreducible matrix explore the whole positive orthant if we tune the matrix's non-zero entries?"*

Or even as

Problem 3.3 (Full Control (version 3)). *Let $M_S(\mathbb{R})$ be the space of non-negative,*

[1] In this chapter we are going to be talking about a notion of controllability which is less known within the complex systems literature. This is "static" controllability, as opposed to dynamic controllability [76], because we are not considering dynamical processes in the network whose behavior we want to adjust.

irreducible matrices with zero entries at fixed indices $S = \{(i,j) \mid M_{ij} = 0, \forall M \in M_S\}$, *and let* $\mathcal{P} : M_S \to \mathbf{c}$ *with* $\mathbf{c} > 0$ *be the function which associates each matrix* $M \in M_S$ *to its Perron eigenvector* $\mathbf{c}$. *Under which conditions is* $\mathcal{P}$ *surjective?*

The same applies to many other problems, and actually these shifts in perspective are also essential to solve some of them. An example of this will be studied in Subsection 3.4.2, where we translate different rankings in PageRank to different sections of a simplex, a shift in perspective that sheds light on the problem.

In Section 3.1 we first establish the types of centrality control that we will be discussing. In Section 3.2 we proceed to analyze and discuss those related to structural changes in the eigenvector centrality of standard graphs. In Section 3.3 we discuss structural changes in other measures and networks. We finalize the chapter with the case of parametric control in Section 3.4.

3.1 Classification of control problems

We begin this chapter by establishing some terminology related to the different types of control which can be exerted on centrality measures.

In our study, we assume the role of an omniscient being, which has access to all information available in the systems, i.e. the structure and topology of the network, the centrality measure used, the parameters involved (if any), etc.

3.1.1 Allowed changes: structural vs parametric

The first distinction we are going to introduce separates the actions one could take in order to alter the centrality of a network in two, depending on whether the network or the parameters of the measure are altered.

The first scenario involves structural changes to the topology of the network. Examples of changes which are allowed in this case could be adding nodes or edges to the network, removing them, adjusting the weights of certain edges, changing directions of edges, etc.

The second scenario, altering parameters of the centrality measure, only applies to measures where parameters are involved, evidently. Luckily for us, one of the most prominent spectral centrality measures, PageRank, is an example of a measure where these changes can have impactful results.

It should be noted that, from a practical point of view, these two approaches are completely different. Structural changes are "intrinsic", in the sense that individuals embedded in the network may take advantage of them (for instance, one can decide to establish a new connection with someone in a social network to increase their score), while parametric changes are "extrinsic", in the sense that whoever set the centrality measure in place (e.g. Google, to rank webpages) decides the parameters used, without any regard to the individuals embedded within.

3.1.1.1 On the reach of structural changes: local, quasi-local and global

It is important to understand that, in the case of structural changes, the type of them that we allow have different effects depending on the measure considered, related to how far do changes reach.

Take, for instance, the addition of a single edge in a large network. From the point of view of the degree centrality, this has a "local effect", as only two nodes' scores are affected. From the point of view of the betweenness centrality this will likely have a "quasi-local" effect, as some nodes will benefit from new shortest paths but others might not, if their shortest paths remain the same. From the point of view of the Katz centrality, this change will instead have a "global effect", as it consists of a sum over all possible paths (which are not slightly different).

This distinction will be very important in the analysis of the goals we pursue. Note that parametric changes are by definition global, in terms of reach (they are agnostic to the network).

3.1.2 Control goal: complete vs ranking vs competitors

As we previously mentioned, we assume complete knowledge of the system, and we strive to understand how do "minimal" changes alter to the centrality outcomes. We can broadly classify these alterations in three categories of interest.

The first problem we will consider is one in which we hope to achieve control over each individual centrality score.

Problem 3.4 (Complete/full control). *Given a graph $G = (V, E)$, a centrality measure $f : V \to \mathbb{R}$ and a vector $\mathbf{c} > 0$, can we always modify G (structural change) or f (parametric change) in such a way that the centrality corresponds to $\mathbf{c}$, i.e. $f(i) = c_i$?*

In other words, being able to choose an arbitrary, positive vector $\mathbf{c}$ to be the outcome of the centrality measure on the network, after applying the allowed changes.

However, in most use cases, centralities are used to assess the importance of certain individuals relative to others. In that sense, the actual scores are not relevant, but the relative position between them: who is the most important, who is the second most important, etc. This is referred to as the ranking, and given a graph $G = (V, E)$ with $N = |V|$ nodes it can be defined by means of a bijective function $\phi : V \to \{1, ..., N\}$, where the ranking is given by

$$\phi^{-1}(1) > \phi^{-1}(2) > \cdots > \phi^{-1}(N), \tag{3.1}$$

this can be relaxed to weak inequalities if we allow for ties.

When it comes to centrality vectors, we can make this more precise as an equivalence relation [77]: let $\mathbf{c}_1, \mathbf{c}_2$ be two positive centrality vectors, then

$$\mathbf{c}_1 \sim \mathbf{c}_2 \iff \exists\, \alpha > 0,\ \beta > (-\min_i (\mathbf{c}_1)_i) \text{ such that } \mathbf{c}_1 = \alpha(\mathbf{c}_2 + \beta \mathbf{e}), \tag{3.2}$$

where $\mathbf{e} = (1, \ldots, 1)^T$. This is due to the fact that neither shifting all scores by the same number or scaling them all at once change the overall ranking. Each equivalence class thus corresponds to a different ranking.

Problem 3.5 (Ranking control). *Given a graph $G = (V, E)$, a centrality measure $f : V \to \mathbb{R}$ and a ranking ϕ, can we always modify G (structural change) or f (parametric change) in such a way that the centrality vector's components are sorted according to ϕ?*

In other words for every possible ranking ϕ (without ties there are $N!$ of them), being able to find centrality vectors satisfying

$$c_{\phi^{-1}(1)} > c_{\phi^{-1}(2)} > \cdots > c_{\phi^{-1}(N)}. \tag{3.3}$$

Needless to say, ranking control is guaranteed whenever we have complete control, but not the other way around.

Both the complete and ranking control variants are concerned with tweaking all scores at the same time. However, sometimes we are not interested in tweaking the global centrality vector, sometimes it is sufficient to make certain nodes surpass others in importance [78]. The last problem we will be considering is that of the localization of individual scores, and its relation to effective competitors and leaders.

Problem 3.6 (Localization, effective competitors and leaders). *Given a graph $G = (V, E)$ and a centrality measure $f : V \to \mathbb{R}$ depending on some parameters, we define*

- *The localization a node's centrality to be the set of values it can take upon changing the parameters.*
- *The effective competitors of the graph to be pairs of nodes whose relative position in the centrality ranking is exchanged upon changing the parameters.*
- *The leadership group of the graph to be the set of nodes each of which are ranked first for some value of the parameters.*

It is worth noting that there is one other related problem already discussed in the literature (see e.g. [79]), the so-called "inverse centrality problem". In this case one sets out to prove the existence of a graph with a pre-defined centrality, rather than assuming an underlying one and certain modifications. There are different variants of these problems depending on which kinds of graphs can be obtained (weighted, directed, etc), neither of which will be discussed here as they are out of the scope of this work.

3.2 Structural changes in the eigenvector centrality

We begin our journey through the controllability of centrality measures with the kinds of changes which are structural, or intrinsic. In these scenarios we are allowed to modify the graph G itself. In that regard, there are different changes one can consider, which are the backbone of the following subsections: modifying the weights of existing edges (Subsection 3.2.1), adding multiedges (Subsection 3.2.2), adding self-loops (Subsection 3.2.3) or even choosing/changing directions of existing edges (Subsection 3.2.4).

3.2.1 Weight tuning

As we saw in the previous chapter, there is a measure of the centrality of a network which employs the idea that a node is more central the more central its neighbors are. This measure relies on the computation of the eigenvector of the transposed adjacency matrix corresponding to the spectral radius, and it is a very important quantification of the centrality of each node.

In [24], the authors carried out a complete characterization of the relation between changing the weights of a subset of edges and changing the centrality score of each node.

The most important result in the aforementioned paper is the following theorem:

Theorem 3.7. *Let $G = (V, E)$ be a directed, possibly weighted, strongly connected graph with $N = |V|$ nodes, let $\mathbf{c} \in \mathbb{R}^N$ be a positive vector. It is always possible to assign weights w_{ij} to edges $(i, j) \in E$ such that the eigenvector centrality of G is $\mathbf{c}$.*

This is a very powerful result: no matter the internal connectivity of a directed network, as long as it is strongly connected we can always find some weighting of its edges such that the eigenvector centrality of the weighted graph is one of our choosing. An example of this can be seen in Figure 3.1.

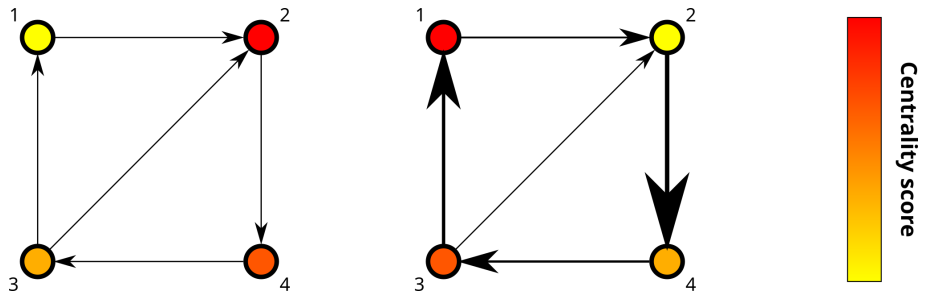


Figure 3.1: A toy example of a directed, strongly connected graph with 4 nodes and 5 edges, whose centrality ranking has been modified to be the opposite of the original one via weight changes, represented as thicker arrows.

Notice that the weights appear linearly in the eigenvector equation. Solving that equation for the weights, a relation between the chosen centrality scores and a

possible, positive weighting (there can be other valid ones) is also given explicitly as

$$w_{i,1} = w_{i,2} = \cdots = w_{i,k_i^{in}} = \frac{\lambda c_i}{\sum_{l=1}^{k_i^{in}} c_{S_{i,l}}}, \tag{3.4}$$

where $S_{i,l}$ is the source node of the l-th arc entering node i, with $l = 1, ..., k_i^{in}$. Notice that this also allows us to establish the value of the largest eigenvalue λ, as if it were a parameter (in the end, multiplying all weights by a common factor only rescales the eigenvalues, while keeping the eigenvectors intact).

Perhaps more interestingly then, is the question of finding the minimum subset of edges whose weights need to be changed for any desired ranking to be achieved. This question was also addressed in [24], but rather than considering the edges whose weights need to be changed, they considered a related problem, which is finding the Minimum Controlling Subset (MCS) of nodes, whose edges need to have different weights. Surprisingly, they found that in many cases of real networks, the relative size of the MCS compared to the size of the full network is quite small, around 10% of it or even less.

Pseudo-Hamiltonian cycles and centrality: here it will be interesting to consider a similar a structure to that of MCS, but this time focused on the minimum number of edges controlling the centrality rather than nodes. This structure is essentially the Hamiltonian cycles, an slight variation of the Hamiltonian paths introduced in Section 2.1. A Hamiltonian cycle is a cycle which passes through every node once and only once before returning to the original node. For our use case, however, we need to allow visiting nodes more than once, so long as no edges are repeated in the process (see graph $\mathcal{H}_G^1$ in Figure for an example). We will refer to such structures as *pseudo-Hamiltonian cycles*.

If we turn back to centrality, we will show that the subgraph of G spanned by any pseudo-Hamiltonian cycle, which we call $\mathcal{H}_G$ is minimally providing the strong connectivity of G.

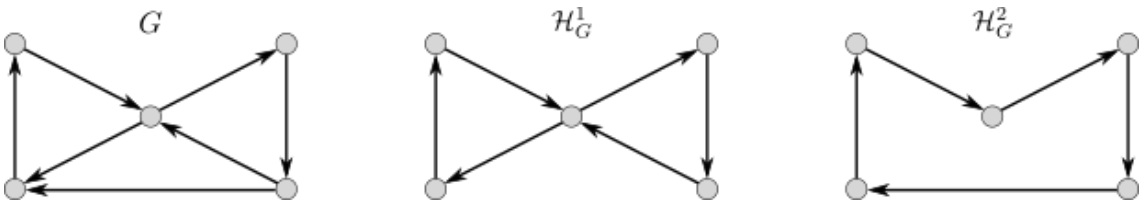


Figure 3.2: An example of a directed graph G with 5 nodes and 7 edges, having two distinct pseudo-Hamiltonian cycles $\mathcal{H}_G$.

Some properties worth noting:

- The graph $\mathcal{H}_G$ need not be unique, as shown in Figure 3.2.
- Clearly, every graph $\mathcal{H}_G$ is strongly connected.

- No edge can be removed from $\mathcal{H}_G$ without destroying its strong connectivity. That is why they need to be cycles (with self intersections allowed) and not just Hamiltonian paths.

This construct allows us to state the following theorem.

Theorem 3.8. *Let $G = (V, E)$ be a directed, weighted, strongly connected graph, where $|V| = N$, and let $\mathcal{H}_G = (V, E_{\mathcal{H}_G})$ be any of its pseudo-Hamiltonian cycles. Let $\mathbf{c} \in \mathbb{R}^N$ be a positive vector. We can always choose a positive weighting on $\mathcal{H}_G$, $W : E_{\mathcal{H}_G} \to \mathbb{R}^+$, such that $\mathbf{c}$ is the eigenvector centrality.*

Proof. We will prove this in two steps.

1. By virtue of Theorem 3.7, any $\mathcal{H}_G$ can be controlled by changing its own weights, as it is strongly connected by definition. If the original graph was already the pseudo-Hamiltonian cycle $G = \mathcal{H}_G$, we are done.

2. If there are extra edges not participating in the pseudo-Hamiltonian cycle. They will have fixed weights f_i (labeled by the node i where the edge ends), which become constant, positive terms in the left-hand side of the eigenvalue equation $A^T\mathbf{c} = \lambda\mathbf{c}$.

   These are subtracted in the right-hand side, for instance the jth equation will be proportional to

   $$\lambda c_j - \sum_{j\to i} f_i c_i. \tag{3.5}$$

   This is problematic, as it can lead to a solution with negative weights. Thus we need

   $$\lambda c_j - \sum_{j\to i} f_i c_i > 0 \Rightarrow \lambda > \frac{1}{c_j}\sum_{j\to i} f_i c_i. \tag{3.6}$$

   Actually, as the following inequality always holds,

   $$\sum_{j\to i} f_i c_i \leq \max_i(f_i)\sum_{j\to i} c_i < \max_i(f_i), \tag{3.7}$$

   we can always set $\lambda = \max_i(f_i)/\min_j(c_j)$.

□

Computational complexity: We've seen the relevance of Hamiltonian cycles in the control of a directed network. But, can we easily find them? That's not an easy question: in fact, quite the opposite. Given a directed graph G, the question of whether it contains Hamiltonian cycles is an NP-complete problem [80].

3.2.1.1 No control in undirected graphs

It is natural to consider whether we can extend the previous result to undirected networks, or if not, understand what is stopping us. The answer is that no, in general there is no control (full or ranking) in undirected networks.

The impossibility of tuning the centralities of undirected graphs changing the weights of their edges can be traced back to the fact that an undirected edge provides the same information (in the sense of its weight) to both ends, therefore we can't expect to create any kind of asymmetry from an intrinsically symmetric object. To make this statement more clear, consider the following another counterexample.

Counterexample 3.9. *Let G be a graph with two nodes and a single edge between them. Clearly, no matter the weight of the edge those nodes will forever have the same centrality.*

3.2.1.2 Control of the original graph via line graph

We know from [36] that the eigenvector centrality of the graph and that of its associated linegraph are related by

$$c_i^G = \sum_{i \to j} c_{i \to j}^{L(G)}. \tag{3.8}$$

We also know from [36] that, if a (directed) graph is (strongly) connected, its associated line graph is (strongly) connected. Therefore, we have the following result.

Theorem 3.10 (Full centrality control tuning the line graph). *Let G be a directed, strongly connected graph with associated line graph $L(G)$. There is full control over the eigenvector centrality of G by suitably choosing the weights of edges in $L(G)$.*

Proof. If G is strongly connected, so is $L(G)$. Therefore, from [24] we know that suitable choices of the weights of $L(G)$ allow us to obtain any desired centrality $c^{L(G)}$.

Consider now the minimal strongly connected graph of N^G nodes: a directed cycle of length N^G. It is clearly isomorphic to its line graph, in particular they have the same number of nodes. Any non-minimal directed, strongly connected graph of N^G nodes will therefore have $N^G < N^{L(G)}$.

It is thus evident from (3.8) that each c_i is the sum of at least one centrality in $L(G)$, and these can be fixed as we consider, so we can fix $\mathbf{c}^G$ accordingly. □

It is important to remark, however, that the resulting centrality $\mathbf{c}^G \in \mathbb{R}^N$ is not necessarily related to any usual notion of eigenvector centrality in G: it has been obtained from equation (3.8), with no reference whatsoever to how the weights relate back from the line graph to the graph itself.

3.2.1.3 No control of the line graph via the original graph

Can we change the weights of the original graph w^G in such a way that we find that we can fix the centrality of the line graph as we desire? This is an interesting question for mainly two reasons:

1. From a practical point of view, sometimes one can model real systems using both the graph and the line graph, with the latter being more relevant but being able to control the former. For instance, in an urban network where crossings are represented by nodes the line graph's nodes represent streets between crossings. Urban planners can choose to improve certain streets/crossings in order to make some streets more relevant or not.

2. From a theoretical point of view, the line graph is the basis of an important set of centrality measures, the "non-backtracking" eigenvector centralities [70, 72] and PageRank [73]. Thus, if we are able to control it then we might have a chance to control those as well.

In Section 2.1 we only defined the line graph for unweighted graphs. In fact, the notion of weighted line graphs is not very well established. Nevertheless, there are different routes one can take in order to translate the graph's weights to the line graph's weights; here we explore two sensible cases, weighted nodes or edges in the original graph.

Weighted edges case. We define the weights of the line graph $w^{L(G)}$ from those on the original graph w^G as

$$w^{L(G)}_{(ij,jk)} = \sqrt{w^G_{ij} w^G_{jk}}. \tag{3.9}$$

The justification for this weight assignment is that it preserves the relation [36] between the adjacency matrices of the graph G, the bipartite graph $B(G)$ and the line graph $L(G)$ given by (2.12). From the point of view of the application to urban systems described above, this could have the interpretation of improving the street itself, hoping that it would become more important.

Indeed, one could expect, at first, that the weight of an edge in the graph and the centrality of said node in the line graph could be related, thus providing complete control over the latter modifying the former. However, as we will see in the following counterexample, this is not the case.

Counterexample 3.11. *Let G be the graph with associated line graph $L(G)$ shown in Figure 2.2. Their respective adjacency matrices are:*

$$A^G = \begin{pmatrix} 0 & a^2 & 0 & 0 \\ 0 & 0 & b^2 & 0 \\ 0 & 0 & 0 & c^2 \\ d^2 & e^2 & 0 & 0 \end{pmatrix}, \quad A^{L(G)} = \begin{pmatrix} 0 & ab & 0 & 0 & 0 \\ 0 & 0 & bc & 0 & 0 \\ 0 & 0 & 0 & cd & ce \\ da & 0 & 0 & 0 & 0 \\ 0 & eb & 0 & 0 & 0 \end{pmatrix}. \tag{3.10}$$

As G is strongly connected, so is $L(G)$, therefore the line graph's eigenvector centrality is well defined, being

$$(A^{L(G)})^T\mathbf{v} = \lambda\mathbf{v} \Rightarrow \begin{cases} (da)v_4 = \lambda v_1, \\ (ab)v_1 + (eb)v_5 = \lambda v_2, \\ (bc)v_2 = \lambda v_3, \\ (cd)v_3 = \lambda v_4, \\ (ce)v_3 = \lambda v_5. \end{cases} \tag{3.11}$$

If we start massaging the equations (assuming that all of its ingredients are non-zero), we end up with conditions like

$$c^2(v_2^2v_4^2 - v_1^2v_3^2) = \lambda v_4^2v_5^2, \tag{3.12}$$

where the left-hand side must be positive. This proves that, for instance, any configuration of centralities $(v_1 = v_2) \wedge (v_4 < v_3)$ is forbidden, so not even ranking control is allowed.

Weighted nodes case. Another alternative to weighting line graphs which is sensible is the case of originally having a node-weighted graph: a graph whose nodes possess a positive numeric value. From the point of view of the application to urban planning described above, this could have the interpretation of improving the crossings between streets.

In this situation we can provide a weight to each line graph edge which is equal to the weight of the node in the original graph which belongs to the endpoint of the two edges connected in the line graph, i.e.

$$w^{L(G)}_{(ij,jk)} = w^G_j. \tag{3.13}$$

Sadly we can also find that controllability is not guaranteed in this setting.

Counterexample 3.12. *Consider the same graph G as in the previous example, with the exception that now the weights are not in the edges of the original graph but in the nodes, i.e.*

$$w^G_1 = a, \quad w^G_2 = b, \quad w^G_3 = c, \quad w^G_4 = d. \tag{3.14}$$

We can then weight the line graph as in equation (3.13). We have

$$A^{L(G)} = \begin{pmatrix} 0 & b & 0 & 0 & 0 \\ 0 & 0 & c & 0 & 0 \\ 0 & 0 & 0 & d & d \\ a & 0 & 0 & 0 & 0 \\ 0 & b & 0 & 0 & 0 \end{pmatrix} \Rightarrow (A^{L(G)})^T\mathbf{v} = \lambda\mathbf{v} \Rightarrow \begin{cases} av_4 = \lambda v_1, \\ bv_1 + bv_5 = \lambda v_2, \\ cv_2 = \lambda v_3, \\ dv_3 = \lambda v_4, \\ dv_3 = \lambda v_5. \end{cases} \tag{3.15}$$

In this case it is straightforward to see that we can't have complete/ranking control, as $v_4 = v_5$.

3.2.2 Multiedge tuning

So far we have discussed graphs which only allowed for one edge per direction between two nodes. But what if we do not want to tweak the weights? We can use multigraphs (graphs allowing for more than a single connection between two nodes), and hope that by tuning the number of edges between two nodes we could adjust the centrality as we wish.

This is a very realistic situation in many scenarios. For instance, in the Internet, webpages can link to one another with hyperlinks, which do not have weight, but there can be several of these hyperlinks stacked together. The same goes for road networks; usually roads can be expanded to allow for different lanes, thus increasing the cardinality of that edge.

To start things off, this is does not provide complete control. The simplest way to disprove it is by using a cardinality argument: centrality scores are real numbers, but the number of in- and out-edges are integers (i.e. countable infinity vs uncountable infinity). There is no way in which adding/removing edges can account for any possible centrality vector.

However, we can still retain ranking control, as we now turn to show.

Theorem 3.13 (Ranking control with multiedges). *Let $G = (V, E, W)$ be a multigraph, where V is the set of nodes, E is the set of edges and $W : E \to \mathbb{N}$ is the number of edges between two nodes. If we are allowed to modify W as we want, we can achieve any possible ranking $c_{i_1} < c_{i_2} < \cdots < c_{i_n}$ with $(i_1, i_2, \ldots i_n)$ any ordering the nodes.*

Proof. We can prove this in three steps.

1. Without loss of generality, we choose a desired ranking given by integer-valued centrality scores, i.e. $\mathbf{c}$ such that $c_{i_1} < ... < c_{i_n}$, $c_i \in \mathbb{N}$.

2. As we discussed in Subsection 3.2.1, we can control the centrality of G changing the weights of the network [24], therefore we obtain the weighted graph $G^{(w)}$ associated with the previous ranking, with adjacency matrix $A^{(w)}$.

   A possible set of weights accounting for this ranking are given by equation (3.4). As the centralities are integer-valued, if we also set the eigenvalue in that equation to be an integer (which we can [24]), then we ensure that all weights are rational, in other words $A^{(w)} \in \mathbb{Q}^{N\times N}$.

3. We also know that scaling a matrix by a factor $\Omega \in \mathbb{R}\backslash\{0\}$ only scales the eigenvalues, while eigenvectors remain the same. Therefore, we can choose Ω arbitrarily large, and multiply it times the weighted adjacency matrix $\Omega A^{(w)} \equiv A^{(\Omega w)} \in \mathbb{N}^{N\times N}$ to obtain another graph weighted $G^{(\Omega w)}$ with the exact same ranking as before but integer-valued weights (i.e. multiedges).

   The choice of Ω is then dictated by the maximum common divisor of the

weights, or, to make things simpler

$$\Omega = \prod_{i=1}^{N} \left(\sum_{l=1}^{k_i^{in}} c_{S_{i,l}} \right) \in \mathbb{N}, \tag{3.16}$$

which completes the proof.

□

3.2.3 Self-loop tuning and competitors

Here we will consider how do self-loops affect the eigenvector centrality of a network, in different scenarios. We will initially investigate the case of a node adding self-loops to itself, to then move on to a complete control over all self-loops of the network.

3.2.3.1 Raising a single node's importance

This type of centrality modification aims to allow a node to become the most important one by increasing its self-references (self-loops). In the classification of control problems established at the beginning of this chapter, this would fall into the localization and competitors category, where the competitors of the node with the self-loop are all other nodes in the network.

We can confirm our intuition about this being a plausible way to become the most central, with the following result.

Theorem 3.14 (Self-loop eigenvector centrality increase). *Let $G = (V, E)$ be a graph. Increasing the weight of a node's self-loop will at some point make that node the most central with respect to the eigenvector centrality.*

Proof. Without loss of generality, we choose node 1 to be that with a tunable self-loop weight, which we want to increase. This tuning is tantamount to adding the matrix $\mathbf{e}_1 \cdot \mathbf{e}_1^T$ enough times to the original adjacency matrix.

However, it is better to think of it in reverse: we add to this matrix the adjacency matrix times an arbitrary small parameter ϵ, i.e. we consider that the network structure is subleading compared to the self-loop.

$$A(\epsilon) = \mathbf{e}_1 \cdot \mathbf{e}_1^T + \epsilon A, \quad A^T(\epsilon)\mathbf{c}(\epsilon) = \lambda_1(\epsilon)\mathbf{c}(\epsilon). \tag{3.17}$$

Let us recall that the eigenvalues of a matrix depending on a real parameter $t \in \mathbb{R}$ are continuous functions $\lambda_1(t), \lambda_2(t), ..., \lambda_N(t)$ [81]. Therefore we know that the spectral radius $\rho(A^T(\epsilon)) = \lambda_1(\epsilon)$ satisfies $\lim_{\epsilon \to 0} \lambda_1(\epsilon) = 1$.

Due to the non-negativity of we have $\lambda_1(\epsilon) = 1 + \mathcal{O}(\epsilon)$. On the other hand, we have $\mathbf{c}(\epsilon) = c_1(\epsilon)\mathbf{e}_1 + c_2(\epsilon)\mathbf{e}_2 + ... + c_N(\epsilon)\mathbf{e}_N$. Inserting both expressions in (3.17) we find

$$A^T(\epsilon) \left[c_1(\epsilon)\mathbf{e}_1 + ... + c_N(\epsilon)\mathbf{e}_N \right] = (1 + \mathcal{O}(\epsilon)) \left[c_1(\epsilon)\mathbf{e}_1 + ... + c_N(\epsilon)\mathbf{e}_N \right] \tag{3.18}$$

From this we conclude that the only term which can be finite is that of $c_1(\epsilon) = \mathcal{O}(1)$, while the rest need to be subleading $c_i = \mathcal{O}(\epsilon)$, $\forall i \neq 1$. □

3.2.3.2 Tuning all self-loops

It is not very surprising that if we are allowed to choose the weight of each and every self-loop in a directed network, we can always set the centrality of a network (directed or undirected, weighted or unweighted) as we like. In fact, the previous proof was more convoluted than the following one, which also gives us a precise characterization of the self-loops' weights.

Theorem 3.15. *Let $G = (V, E)$ be a (strongly) connected graph and adjacency matrix $A = (a_{ij})$. Let $\mathbf{c}$ be a positive vector. We can always set the weight of the self-loops a_{ii} such that the eigenvector centrality of the network is given by $\mathbf{c}$.*

Proof. The eigenvector centrality equation for node i is given by

$$\lambda c_i = a_{ii}c_i + \sum_{j\neq i} a_{ji}c_j \Rightarrow a_{ii} = \lambda - \frac{1}{c_i}\sum_{j\neq i} a_{ji}c_j, \tag{3.19}$$

where we used the fact that $c_i > 0$. Note that the Perron eigenvalue is a free parameter in the system of equations for all a_{ii}. Therefore, we can set it as well. Choosing it such that

$$\lambda > \max_i \left(\frac{1}{c_i}\sum_{j\neq i} a_{ji}c_j\right) \tag{3.20}$$

will allow us to completely solve for positive self-loop weights. □

Notice that if we are only interested in ranking control, then we can define a notion of controllability for the network based on self-loop weighting by choosing the smallest possible λ needed to yield any ranking.

Definition 3.16 (Self-loop controllability). *Let $G = (V, E)$ be a (strongly) connected graph with N nodes and adjacency matrix $A = (a_{ij})$. Let $\boldsymbol{v} = (1, 2, ..., N)^T \in \mathbb{R}^N$, and $\sigma(\boldsymbol{v})$ the set of $N!$ vectors obtained by permuting the components of $\boldsymbol{v}$. The self-loop controllability $\mathcal{C}$ of G is defined as*

$$\mathcal{C} = 1 - \frac{1}{N}\max_{\mathbf{c}\in\sigma(\boldsymbol{v})}\left[\max_i\left(\frac{1}{c_i}\sum_{j\neq i} a_{ji}c_j\right)\right]. \tag{3.21}$$

The factor $1/N$ is a normalization for the purpose of providing a meaningful comparison when the sizes of the networks considered are the same.

In practice, sampling over all possible rankings is computationally unfeasible, $\mathcal{O}(N!)$. We can sidestep this problem using a powerful idea, which will also be used in Subsection 3.4.2 [67]: obtaining the λ associated to the "equal ranking" centrality, i.e. $\mathbf{c} = (1, ..., 1)^T$, as we know that small variations of the a_{ii} weights will shift it to our desired ranking. This will give an approximate, simplified controllability index $\widehat{\mathcal{C}}$, which actually corresponds to the maximum in-degree (without self-loops) of G:

$$\widehat{\mathcal{C}} = 1 - \frac{1}{N} \max_i \left(\frac{1}{1} \sum_{j \neq i} a_{ji} \cdot 1 \right) = 1 - \frac{1}{N} \max_i (k_i^{\text{in}}) \in \left[\frac{1}{N}, \frac{N-1}{N} \right], \tag{3.22}$$

where the extremal values correspond to a complete network ($\widehat{\mathcal{C}} = \frac{1}{N}$) and a directed ring ($\widehat{\mathcal{C}} = \frac{N-1}{N}$).

The interpretation of this result is the following: given two networks with equal number of nodes, the one with the highest $\widehat{\mathcal{C}}$ (i.e. the one with the highest maximum in-degree) will be more controllable, as we will need a smaller spectral radius λ to provide positive self-loop weights, hence weaker self-loops would be needed to achieve complete ranking control. We can see this coefficient calculated for some real networks in Subsection 3.2.5.

Lastly, let us comment on the difference between this result (tuning all self-loops), compared with the previous one (raising a single one). Clearly, the former is a much stronger result, that furthermore provides us with a specific formula for the self-loop weights. However, the latter is more realistic: usually, a node has access to itself and its surroundings (e.g. a webpage could link itself almost endlessly, but it does not have control over other webpage's self-references), which is why that result is still relevant.

3.2.4 Directionality tuning

Let G be an undirected, connected graph containing at least one cycle. Suppose we have the freedom to turn it into a directed graph, picking the direction(s) in each edge. Can we set the eigenvector centrality or the ranking of the network?

The answer to the former is no (through a cardinality argument identical to that discussed in Subsection 3.2.2). As for the ranking control, we will show that it is not impossible, but nearly so.

3.2.4.1 Total directedness

In this case we can choose only one direction per edge, in other words, each one would only be traversable in one direction. This is way more restrictive, in part due to the fact that strong connectedness needs to be assured for a well-behaved centrality. Let's examine some simple cases.

Complete graph K_N: To begin, let us show that this task is not doomed from the start by discussing the complete graph case, where there actually always is ranking control if $N > 3$ when we direct its edges adequately (resulting in K_N^D).

To see this, notice that the complete graph is very special in the sense that it has total permutation symmetry, therefore we just need to prove that there exists a configuration of the directions such that all centrality scores are different. Then, any other ranking can be obtained as a permutation of the original labelling.

To design a ranking for the complete graph, consider the following wiring scheme.

1. Start with the following complete, oriented graph of $N = 4$ nodes

$$K_4^D : \quad 1 \to 2,\ 2 \to 3,\ 3 \to 4,\ 4 \to 1,\ 3 \to 1,\ 4 \to 2. \tag{3.23}$$

   This graph is clearly strongly connected.

2. Add a single node $N+1$, with connections $4 \to N+1$, $N+1 \to 1$. This already guarantees strong connectedness.

3. The remaining pairs of edges, orient them from node $N+1$ to 2, 3, ... $N-1$.

4. Repeat steps 2-3 until the complete, oriented graph with the desired number of nodes K_N^D is reached.

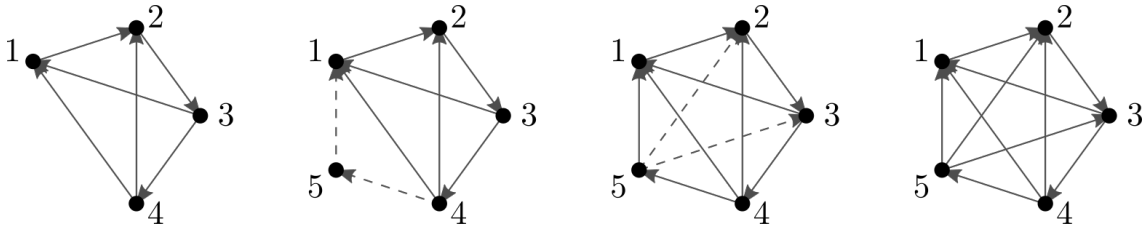


Figure 3.3: Example of construction of K_5^D from K_4^D, following steps 2 and 3.

We can show the following result from this algorithm.

Theorem 3.17. *The previous algorithm yields a sequence of centralities*

$$c_2 > c_1 > c_3 > \cdots > c_{N-1} > c_N \tag{3.24}$$

Proof. First, consider the final adjacency matrix of a directed, strongly connected graph, $A = (a_{ij})$. By the Perron-Frobenius theorem we know that $\rho(A) > \min_i \sum_j a_{ij} = 1$, where the last equality is due to strong connectedness (which enforces all nodes to have an out-degree of at least 1).

Then, from the eigenvector equation of A^T, we know that the contribution to the centrality score of node i from node j is given of order $\mathcal{O}(\rho^{-d(j,i)})$, where $d(j,i)$ is the length of the shortest path from j to i.

Let us now consider how are centralities changing as we follow the algorithm.

1. Graph K_4^D has a Perron eigenvector with $c_2 > c_1 > c_3 > c_4$.

2. The new node will have a direct contribution just from node 4, which was the least important one. Furthermore, as $\rho < 1$ we necessarily obtain $c_4 > c_5$. On the other hand, the centrality of node 1 obtains an increase of order $\mathcal{O}(\rho^{-1})$.

3. The remaining connections also provide a first order $\mathcal{O}(\rho^{-1})$ increase to nodes 2, 3, ... $N-1$ (not only that, but the exact same increase each of them as well as the one that node 1 received), and an increase of order $\mathcal{O}(\rho^{-2})$ to node N.

4. Steps 2-3 would be repeated with the same consequences.

The overall change in centralities after each step is then the following:

$$\Delta c_1 = \frac{c_N}{\rho} + \mathcal{O}(\rho^{-2}), \quad \Delta c_2 = \frac{c_N}{\rho} + \mathcal{O}(\rho^{-2}), \ ..., \ \Delta c_N = \frac{c_{N+1}}{\rho^2} + \mathcal{O}(\rho^{-3}), \tag{3.25}$$

while the centrality of the added node goes as

$$c_{N+1} = \frac{c_N}{\rho} + \mathcal{O}(\rho^{-2}). \tag{3.26}$$

Therefore at each step the sequence stated in the Theorem is preserved. □

Cycle graph C_N: Here the strong connectedness requirement leaves us with essentially no freedom. We need to establish a (counter)clockwise direction in the cycle, and that would be it. Hence, there is absolutely no chance for centrality control in this scenario.

Bound based on the graph density: We can also derive a simple argument against control by total orientation by counting possibilities. Let N be the number of nodes, L be the number of edges. Then, the number of rankings to reach is $N!$, and the possible orientation combinations is 2^L, which in turn implies that $L > \log_2(N!)$ in order to attempt to have ranking control.

For dense graphs this bound does not mean much. For sparse graphs $L \sim \mathcal{O}(N)$ this bound is dangerous: using Stirling, one can derive $\log_2(N!) \sim \mathcal{O}(N \log(N))$, which is greater.

Improving the bound: In order to have strong connectedness, there must be a strongly connected skeleton, as defined in Subsection 3.2.1. Hence at least $L = N$ edges need to have a fixed direction as a directed cycle, therefore once we choose one of either directions, we are only allowed to choose the remaining $L - N$ edge's directions:

$$2(2^{L-N}) > N! \quad \Rightarrow \quad L > N + 1 + \log_2(N!), \tag{3.27}$$

which is a stricter bound. In Subsection 3.2.5 this bound is computed for several real networks.

Subgraphs within general networks: It is sort of intuitive that there will be a lack of ranking control with orientation choices. The way to formalize this idea is finding patterns which defy ranking control (similar to the C_n case) within graphs.

- The simplest one is finding a "bridging edge": an edge (i, j) such that $k_i = k_j = 2$ (prior to directing it). Strong connectedness requires both nodes to have an in-edge and an out-edge. Choosing, without loss of generality, $n \to i \to j \to m$ (where n, m are the two contiguous nodes), we have that centrality of both nodes as well as that of the predecessor one are related by $c_j = c_i\rho = c_{pred}/\rho$ with ρ the spectral radius. By Perron-Frobenius, the spectral radius is $\min_i \sum_j a_{ij} \leq \rho$, hence $\rho \geq 1$ and therefore. $c_n \geq c_i \geq c_j$. It is clear that it will not be possible for us to obtain an inequality $c_j \geq c_i \geq c_n$ (unless G is a cycle and the equality holds, although there is still no ranking control), hence that ranking is possible, and therefore there is no ranking control.
- Cycles are very dangerous for ranking control:
  - If there is a cycle which is only connected to the bulk of the graph by one of its nodes, the above reasoning for C_n applies.
  - If it is connected to the bulk by two different nodes, then we can divide the cycle in two paths, and check if either contains bridging edges (for C_5 and higher this is always the case).
  - The previous point can be extended to more connections: k connections avoid bridging edges if $n \geq 2k$, therefore a cycle C_n with $n > 2k$ will always destroy ranking control.

Lack of ranking inversion: If all orientations are reversed, is the ranking also reversed? That would actually allow us to only need to prove that half as many rankings could be achieved, with the latter half recovered by inversion. But that is not the case, the inversion has a non-trivial effect: take as an example the complete graph K_4. By Theorem 3.17 we know that following the algorithm we obtain the ranking $c_2 > c_1 > c_3 > c_4$. If we reverse each and every orientation, again using Theorem 3.17 we would obtain $c_3 > c_4 > c_2 > c_1$, hence they are not inverted.

3.2.4.2 Partial directedness

In this case we can choose to establish either a single direction or both directions per edge. This is a far harder question, as we can no longer rely on the constraint of strong connectedness: if the undirected graph is connected we could always assign both directions when orienting it.

Bound based on the graph density: This bound is relaxed but still imposes some strict constraints. We now have three possible choice for each edge, therefore

$$3^L > N! \quad \Rightarrow \quad L > \log_3(N!), \tag{3.28}$$

however the lack of tighter restraints disables further improvements of this bound. Still, it is a death sentence for ranking control in sparse graphs, as we will see in what follows.

3.2.5 Numerical estimates of some of the previous schemes

Here we gather some of the previous findings related to different types of network controllability, applying them to real network datasets. These datasets are all publicly available in the CASOS network repository [46].

Network	N	L	LCC %	$\widehat{\mathcal{C}}$ (3.22)	Total (3.27)	Partial (3.28)
a2c2_team_d_no_scuds	144	264	59.75%	0.667	No	No
a2c2_team_d_scuds	166	291	65.87%	0.705	No	No
bkfrat	58	1653	100%	0.017	Yes	Yes
company	59	191	100%	0.576	No	Yes
dolphins	62	159	100%	0.806	No	No
football	115	613	100%	0.896	No	Yes
krebs_anytie	73	2628	100%	0.014	Yes	Yes
polbooks	105	441	100%	0.762	No	Yes
sg1	57	152	95.0%	0.491	No	No
stargate	71	252	100%	0.592	No	Yes
tanzania-kenya-imoon	75	357	100%	0.613	No	Yes
tanzania_3_2009b	151	555	94.38%	0.497	No	No
tanzania_3_2009b_trails	149	484	93.12%	0.624	No	No

Table 3.1: Statistics of real networks with respect to some of the controllability measures and bounds described previously, namely: number of nodes, number of edges, percentage of nodes in the largest connected component, self-loop controllability index, and whether the bounds of total and partial directedness are satisfied.

We can observe clearly in Table 3.1 that total controllability is essentially impossible in these networks. As a matter of fact, the two which are totally controllable are so because they are actually complete graphs. This is also reflected in the low values of the self-loop controllability index $\widehat{\mathcal{C}}$ (reaching the lower bound).

This is even more evident if we plot these results as in Figure 3.4. In that regard, it is interesting to note that the self-controllability index is not very correlated with the total/partial controllability, except for the extreme cases of the complete networks.

3.3 Structural changes in other spectral measures

The previous section was devoted to understanding the effect of different structural changes in the outcomes of the eigenvector centrality. Before moving on to parametric controllability, we want to spend some time discussing other centrality measures and their structural controllability, or lack thereof.

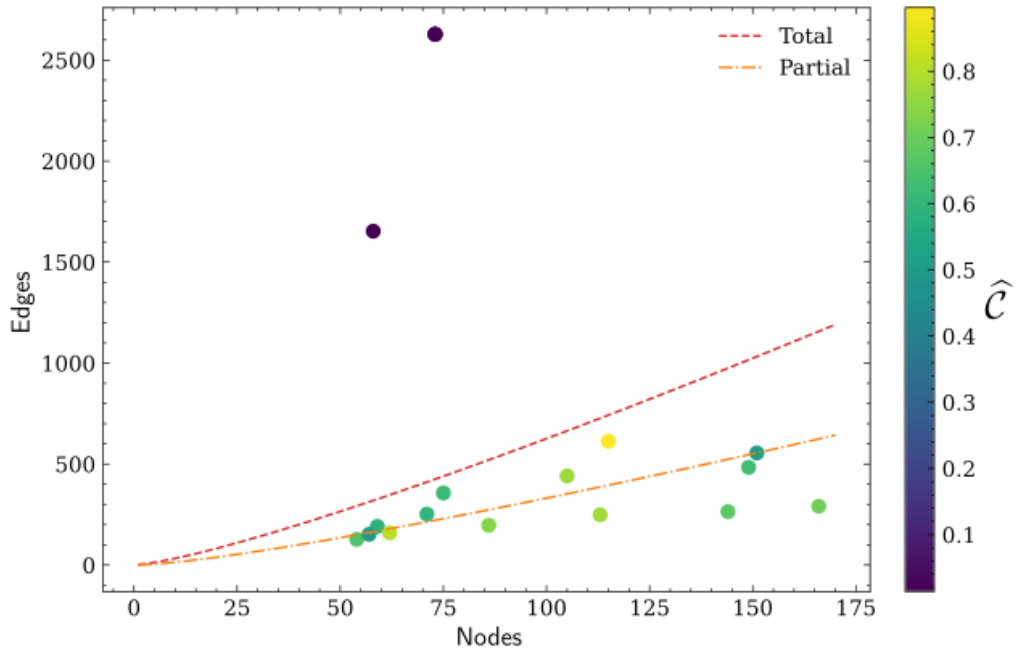


Figure 3.4: Scatter plot showing the number of edges against the number of nodes for the networks in Table 3.1, with datapoints colored based on $\widehat{\mathcal{C}}$.

3.3.1 Weight tuning in non-backtracking centralities

As for the controllability of this measure via weight tuning of the original network, sadly we are at a loss. The reason for this was already advanced in Subsection 3.2.1: the Hashimoto matrix is intrinsically connected to the line graph's adjacency matrix, and we showed that its centralities can't be successfully controlled from tuning the weights of the original graph.

3.3.2 Weight tuning in PageRank

The deep relation between the eigenvector centrality and PageRank, mediated by the Perron-Frobenius Theorem A.2 make it quite tantalizing the possibility of applying the same method which worked in the directed graph case to PageRank. Sadly, as we will clearly see there is a fundamental problem in this scenario, which renders PageRank generally uncontrollable using this method even in the directed graph case.

The reason for this is the row-normalization of the adjacency matrix: the construction of P normalizes out any weight placed on out-edges coming from nodes with out-degree equal to 1. The simplest way to see this is considering directed rings, as in the following counterexample.

Counterexample 3.18. *Consider the $N = 6$ directed cycle $G = (V, E)$ from Figure 3.5. Its adjacency matrix is of the form $a_{ij} = \delta_{j,(i+1 \mod N)}$. Each node clearly has $k_i^{\text{in}} = k_i^{\text{out}} = 1$, $\forall i$. If we modify the weight of edge $(i, j) \in E$ to be $w_{ij} > 0$, then*

$k_i^{out} = w_{ij}$ *and it therefore translates in the row-adjacency matrix* P *of PageRank* (2.25) *back to* $p_{ij} = 1$*, regardless of* w_{ij}*.*

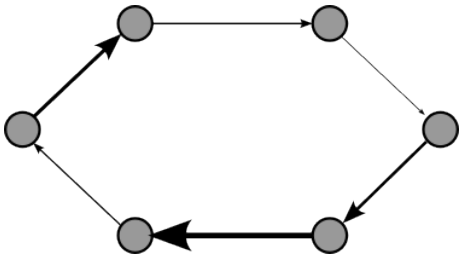


Figure 3.5: Simple example of a network (the directed cycle C_6) whose PageRank centrality is unaffected by any modification of the edge weights.

3.3.3 Self-loop tuning in PageRank

In the previous Section we found that it is always possible to increase a node's own centrality with enough self-loop weight over it. The PageRank case is more nuanced: increasing the self-loop weight will essentially diminish the weights of other edges, coming out of that node, but it will not affect (comparatively) the rest of the network. This makes that strategy less viable. Still, we can derive some interesting bounds for when it does work.

Before doing so, let us enunciate the following important result

Theorem 3.19. *(Theorem 8, p. 130 in [82]) Let* $A(t)$ *be a differentiable matrix-valued function of* t*,* $a(t)$ *an eigenvalue of* $A(t)$ *of multiplicity one. Then we can choose an eigenvector* $\boldsymbol{h}(t)$ *of* $A(t)$ *pertaining to the eigenvalue* $a(t)$ *to depend differentiably on* t*.*

This will be a key result in the upcoming theorem, which shows that a node can increase its centrality score beyond that of the rest of the nodes after increasing its self-weight and the damping factor high enough.

Theorem 3.20 (Self-loop PageRank increase). *Let* $G = (V, E)$ *be a strongly connected graph, with row-normalized adjacency matrix* P*. If we allow a node to establish an arbitrary number of self-loops onto itself (alternatively, a single self-loop with arbitrarily high weight), then for* $1 - 1/w \leq \alpha \leq 1$ *that node can always achieve the highest PageRank value.*

Proof. Without loss of generality, we choose node 1 to be that with a tunable self-loop weight. For simplicity, we first consider the $\alpha = 1$ (no personalization) case, we later discuss what changes when $\alpha \neq 0$.

In that scenario, we need to compute the Perron eigenvector of P. The row normalization yields the following values

$$P_{11} = \frac{w}{k_1^{out} + w}, \quad P_{1n} = \frac{a_{1n}}{k_1^{out} + w} \quad \Rightarrow \quad P(w) = \left(\begin{array}{c|c} \frac{w}{k_1^{out} + w} & \frac{a_{1n}}{k_1^{out} + w} \\ \hline \mathcal{O}(1) & \mathcal{O}(1) \end{array}\right), \tag{3.29}$$

where w is the weight of the self-loop.

Due to the strong connectedness, for finite w we have that $P(w)$ is an irreducible matrix, therefore it has a unique, positive eigenvector $\boldsymbol{\pi}$ associated to the spectral radius $\lambda = 1$.

In the limit we have

$$P_\infty = \lim_{w\to\infty} P(w) \Rightarrow P_\infty^T = \left(\begin{array}{c|c} 1 & \mathcal{O}(1) \\ \hline \mathbf{0} & \mathcal{O}(1) \end{array}\right) \equiv \left(\begin{array}{c|c} 1 & \mathbf{c} \\ \hline \mathbf{0} & B \end{array}\right), \tag{3.30}$$

where the respective submatrices are

$$\mathbf{0} = \begin{pmatrix} 0 \\ \vdots \\ 0 \end{pmatrix} \in \mathbb{R}^{(N-1)\times 1}, \ \mathbf{c} \in \mathbb{R}^{1\times(N-1)}, \ B \in \mathbb{R}^{(N-1)\times(N-1)}. \tag{3.31}$$

P_∞^T possesses $\mathbf{e}_1 = (1, 0, \ldots, 0)^T$ as eigenvector with eigenvalue $\lambda = 1$. The remaining eigenvectors are orthogonal to this one, and their non-trivial part comes from those of the submatrix $B = (b_{ij}) \in \mathbb{R}^{(N-1)\times(N-1)}$. This submatrix is irreducible, and thus the Perron-Frobenius Theorem A.2 applies to it, in particular we have the following property for its spectral radius r_B [33]

$$r_B \leq \max_i \sum_j b_{ij}, \tag{3.32}$$

and due to the fact that B is no longer column stochastic (as was P_∞^T), we have $r_B < 1$. Therefore, $\lambda = 1$ is an eigenvalue of $P(w)$ with algebraic multiplicity 1 for all $w \in (0, \infty)$. Furthermore, $P(w)$ is differentiable in that range, hence by Theorem 3.19 its associated eigenvector is continuous, and therefore the only non-negligible component of the eigenvector for high enough, finite w will be that of the first node.

Lastly, we should consider the effect of decreasing α to less than 1. In that case a sufficient (but not necessary) condition for the first node to still retain the highest importance would be if $1 - 1/w \leq \alpha$. To see this, notice that the Google matrix (2.28) becomes

$$\mathbb{G} = \alpha P + (1 - \alpha)\, e \cdot v^T \leq P + \mathcal{O}(w^{-1})(P + e \cdot v^T). \tag{3.33}$$

Therefore, the leading order contribution is still that coming from P, and the previous arguments still apply. □

3.3.4 Weight tuning in multiplex networks

Before ending the structural control part, we want to have a small excursion around the case of multiplex networks, and their control. Consider the eigenvector-like centrality measures introduced in Subsection 2.2.3. Firstly, we are going to restrict ourselves to the local-heterogeneous and global-heterogeneous ones. The reason for this is rather simple: the other two (independent layer and projection) essentially ignore the interconnectivity of layers, and can be analyzed in the same way as we have done in Subsection 3.2.1 for the single layer case.

3.3.4.1 Undirected uncontrollability

Let us get this out of the way: there is no control in any of the cases, as in the end everything is reduced to computing eigenvector centralities of certain undirected networks obtained from the multiplex, and as we saw in Section 3.2 it is fundamentally impossible to use weight-tuning to control a general undirected network's eigenvector centrality.

3.3.4.2 General uncontrollability

In the directed multiplex case, even if the existence and uniqueness requirements are met and we allow changes to weights, in general we still can't control all the centrality scores as we did previously. To see this, consider the following simple example:

Counterexample 3.21. *Let $\mathcal{G} = (G_1, G_2, G_3)$ be a multiplex network where G_1 is strongly connected, while G_2, G_3 are completely disconnected. Suppose there are identical cross influences $q_{12} = q_{13} = q_{31} = q_{21}$.*

By just staring at the symmetry of this problem we realize that, as layers $2, 3$ are indistinguishable, we shouldn't expect that they could be assigned different centrality vectors.

It is therefore clear that we need to restrict the problem in order to achieve some kind of control. Let us go step by step and see what other problems we can encounter and which situations actually lead to controllability.

3.3.4.3 What can a layer control?

The most basic (and rather trivial) example of structural control in multiplex networks, is that of a single layer controlling some layer's centralities. Before showing the actual theorem, let us show something more general, related to the control of a single layer, which will be useful for us.

Lemma 3.22 (Full control under linear weighting). *Let $G = (V, E)$ be a directed, strongly connected graph where $|V| = N$. Let $\mathbf{c} \in \mathbb{R}^N$ be a positive vector and $\alpha > 0$, $\beta_{ij} \geq 0$. We can always choose a positive weighting $W : E \to \mathbb{R}_+$ such*

that the transpose of the linearly weighted adjacency matrix

$$A^{\text{lin}} = (a_{ij}^{\text{lin}}) = \begin{cases} \alpha w_{ij} + \beta_{ij} & \text{if } (i,j) \in E \\ \beta_{ij} & \text{otherwise,} \end{cases}, \tag{3.34}$$

has $\mathbf{c}$ *as its Perron eigenvector.*

Proof. Consider the graph G^{lin} whose adjacency matrix is A^{lin}. By construction it is strongly connected, hence it contains at least one pseudo-Hamiltonian cycle $\mathcal{H}$. Furthermore, we can find it within the edges of G.

Using Theorems 3.7 and 3.8 we can guarantee the existence of positive weights Ω_{ij} with $(i,j) \in \mathcal{H}$ in the graph G^{lin} such that its eigenvector centrality is $\mathbf{c}$. We can also assume $\Omega_{ij} > \beta_{ij}$, as we can always choose a higher value of the spectral radius. From there it is straightforward to compute the associated values $w_{ij} = \frac{1}{\alpha}(\Omega_{ij} - \beta_{ij}) > 0$. □

This result is important because in the local-heterogeneous centrality case, the adjacency matrices $A_m^\star$ have this form, once we weight non-overlapping edges across layers. Henceforth, we can now tackle our first theorem on multiplex control.

Theorem 3.23 (Full control of a single layer's centralities from another layer). *Let* $\mathcal{G}$ *be a multiplex with* M *layers* $G_m = (V, E_m)$*, where* $|V| = N$*. Let* $\mathbf{c} \in \mathbb{R}^N$ *be a positive vector. We can always choose a positive weighting,* $W_s : E_s \to \mathbb{R}_+$ *on layer* s *such that* $\mathbf{c}$ *is the local-heterogeneous eigenvector-like centrality of layer* s'*.*

Proof. Without loss of generality, let us choose layer $s = 1$ to be the controlling layer, layer $s' = 2$ the layer whose centrality we want to fix. The local-heterogeneous eigenvector-like centrality in layer 2 takes the form $(A_2^\star)^t \mathbf{c}^{(2)} = \lambda_2^\star \mathbf{c}^{(2)}$. The weighting of layer 1 is introduced in $A_2^\star$ in a linear form (where $\alpha \equiv q_{12}$ and β_{ij} accounts for the contribution of layers other than s), hence we can use Lemma 3.22 to guarantee the existence of a positive weighting with $\mathbf{c}^{(2)} = \mathbf{c}$. □

In this case, however, it is not trivial to construct one of such solutions in all generality, as was the case in the standard directed network case, see equation (3.4).

3.3.4.4 What can a pseudo-Hamiltonian cycle, spanning several layers, control?

The above result assumes that there is at least one layer which is strongly connected. However, we know that the local-heterogeneous eigenvector-like centrality of any layer is guaranteed if the projection network is strongly connected [74], regardless of the individual connectivity of each layer.

In that regard, it is easy to note that there is really nothing special about the fact that the weighted edges lied in a single layer. The only requirement is actually finding a pseudo-Hamiltonian cycle (see Subsection 3.2.1 for the definition) in the projection layer (which is already guaranteed if the centrality measure is well-defined).

Definition 3.24 (Projection map). *Let $\mathcal{G}$ be a multiplex with M layers $G_m = (V, E_m)$ and let $\overline{G} = (V, \overline{E})$ be its projection multigraph (where multiedges come from overlapping edges in different layers). We define the projection map $\mathcal{P} : E_m \to \overline{E}$ as the map from an edge in the multiplex to its equivalent edge in the projection multigraph.*

Notice that the reason why we consider a projection *multigraph* allows us to assert that $\mathcal{P}$ is a bijective map, hence $\mathcal{P}^{-1}$ brings edges in the projection back to their original edges in the multiplex.

Corollary 3.25 (Full control of a single layer's centralities with several edges from the multiplex). *Let $\mathcal{G}$ be a multiplex with M layers $G_m = (V, E_m)$, with N nodes. Let $\mathbf{c} \in \mathbb{R}^N$ be a positive vector. We can always choose a positive weighting, $W : \overline{E} \to \mathbb{R}_+$ on the projection, with associated weighting $\mathcal{P}^{-1}W$ on the multiplex, such that $\mathbf{c}$ is the local heterogeneous eigenvector-like centrality of layer s.*

Proof. This follows immediately from Lemma 3.22, where instead of starting from a graph G we start from a pseudo-Hamiltonian cycle in the projection multigraph. □

3.3.4.5 More problems with more control attempts

Given the previous successes, it would be tempting to think that the straightforward generalization of at least Theorem 3.23 to two layers controlling other two centralities would be feasible. However, there are bad news in this front.

Notes on further controllability constraints: Suppose that we have a biplex, with adjacency matrices A_1, A_2. It would not be unreasonable to think that if we can control both layers' weights, the centrality of each layer could be controlled. However this turns out to be more nuanced, due to the cross-influence matrix Q.

Recall that, if two linear operators A, B commute and $\mathbf{c}$ is an eigenvector of A whose eigenvalue λ has multiplicity 1, then $\mathbf{c}$ is an eigenvector of B:

$$AB\mathbf{c} = BA\mathbf{c} = \lambda B\mathbf{c} \Rightarrow B\mathbf{c} \text{ eigenvector of } A \Rightarrow B\mathbf{c} \propto \mathbf{c}. \tag{3.35}$$

Back to the biplex, we can check whether the two matrices commute, finding

$$\begin{aligned}[A_1^\star, A_2^\star] &= q_{11}q_{22}[A_1, A_2] + q_{12}q_{21}[A_2, A_1] \\ &= (q_{11}q_{22} - q_{12}q_{21})[A_1, A_2] = \det(Q)[A_1, A_2]\end{aligned} \tag{3.36}$$

As the Perron eigenvector has a simple eigenvalue, then we know that whichever the weighting on either layer, if $|Q| = 0$ then the local-heterogeneous eigenvector-like centralities of the biplex will be identical.

The generalization of this constraint to multiplex networks with higher layers is the fact that, if we call $Q_{(l,m)}$ the minor of matrix Q associated to entry (l, m), then we can't have at the same time

$$Q_{(1,m)} = 0 \wedge Q_{(2,m)} = 0 \wedge \cdots \wedge Q_{(M,m)} = 0, \quad \forall m. \tag{3.37}$$

And even if we chose a matrix Q satisfying this conditions (something which is actually very reasonable), we would still need to make sure that the resulting, weighted matrices A_m aren't all commuting at once (although this is a minor inconvenience).

3.3.5 A note on *Search Engine Optimization*

Something which we have not studied is the possibility of adding new links among existing nodes, however this is a very realistic possibility in human-made networks such as the Internet: a webpage can choose which hyperlinks it creates, bridging it to other webpages. As a matter of fact, this problem has been thoroughly studied by the Computer Science community, to the point where its practice has been dubbed *Search Engine Optimization* or SEO [2], for short.

Studies in this area exploded in popularity during the first decade of the 2000's, after PageRank made its appearance and thanks to it Google became the predominant search engine, and before Google moved on to more sophisticated and modified versions of the algorithm (with several link spam filters and using Machine Learning, for instance). Plenty of websites tried to attract as much attention as possible, and therefore needed to be highly positioned in Google's ranking. There were many attempts to mischievously exploit the algorithm via link farms or the so-called "Google bombs" [83, 2], but there was also a vast and genuine attempt to understand thoroughly and quantitatively how did changes affect the PageRank centrality. For the interested reader, we refer them to [84, 85, 86, 87, 88] and the references therein.

3.4 Parametric changes

As we discussed at the beginning of this chapter, when it comes to the controllability of centrality measures in networks there is sometimes the possibility of modifying them via the parameters which may be involved in it, if any. Clearly, this is not always the case: for instance, the eigenvector centrality is parameter-free.

However, one of the quintessential centrality measures, PageRank, contains not one, but two parameters: the damping factor $\alpha \in [0, 1]$ and the personalization vector $\boldsymbol{v} \in \mathbb{R}$, $||\boldsymbol{v}||_1 = 1$. We will devote this section to understanding the relation between the centrality outcomes of PageRank (as well as other PageRank-related measures) and the parameters involved, to see how do the latter influence the former.

In order to ease the notation and statements which are to come, we will always be assuming graphs $G = (V, E)$ without dangling nodes (i.e. nodes with zero out-degree, $k_i^{\text{out}} = 0$), for sensibility of the PageRank measure, something we already glossed over in Subsection 2.3.3. If the graph under study were to contain dangling nodes, one can always turn it into one without them by standard methods, e.g. $P \to P + \boldsymbol{d}\boldsymbol{u}^T$ with $\boldsymbol{d} \in \mathbb{R}^N$ the vector indicating dangling nodes and $\boldsymbol{u} \in \mathbb{R}^N, ||\boldsymbol{u}||_1 = 1$ the distribution of dangling nodes.

3.4.1 Localization of PageRank, competitors and leaders

A first step towards the characterization of the relation between centralities and parameters in the PageRank centrality measure can be found in [78], where they associate each node in a network to an interval in the real line which symbolizes the possible values of its centrality score for a fixed damping factor. These intervals are then used to extract information about leaders, followers and competitors in the network. We now briefly review the main points of said article, which we will generalize later on for other PageRank-related measures.

First of all, note that for a fixed graph G and damping factor $\alpha \in (0,1)$ the possible centrality vectors $\boldsymbol{\pi}$ depend on the choice of personalization vector $\boldsymbol{v}$, i.e. $\boldsymbol{\pi} = \boldsymbol{\pi}(\boldsymbol{v})$.

It is natural to then consider the following definition:

Definition 3.26. *Let $G = (V, E)$ be a graph with $|V| = N$. The localization of PageRank for node i is the set of all possible values of the PageRank centrality for said node,*

$$PR(i) = \{\boldsymbol{\pi}^T(\boldsymbol{v})\boldsymbol{e}_i \; \forall \boldsymbol{v} \in \mathbb{R}^N, \boldsymbol{v} > 0, ||\boldsymbol{v}||_1 = 1\}. \tag{3.38}$$

where e_i is the unit vector in the i'th direction. In [78], we find the following theorem.

Theorem 3.27. *Given a graph $G = (V, E)$ and a fixed damping factor $\alpha \in (0,1)$, for each node $i \in \mathbb{N}$*

$$PR(i) = (\min_j x_{ji}, x_{ii}), \tag{3.39}$$

where $X = (x_{ij}) = (1-\alpha)(\mathbb{I}_N - \alpha P)^{-1} \in \mathbb{R}^{N\times N}$.

This is a rather simple but powerful result: one computes matrix X from the row-normalized adjacency matrix of the graph and the damping factor, and its components constrain the possible values of all centrality scores in the graph, regardless of the personalization vector. This information can be used to analyze the existence of "effective competitors" in the network.

Definition 3.28 (Effective competitors). *Given a graph $G = (V, E)$, we say that nodes $i, j \in V$ $(i \neq j)$ are effective competitors if there exist two personalization vectors $\boldsymbol{v}, \boldsymbol{w}$ $(\boldsymbol{v}, \boldsymbol{w} > 0, ||\boldsymbol{v}||_1 = ||\boldsymbol{w}||_1 = 1)$ such that*

$$\boldsymbol{\pi}^T(\boldsymbol{v})\boldsymbol{e}_i > \boldsymbol{\pi}^T(\boldsymbol{v})\boldsymbol{e}_j, \quad \boldsymbol{\pi}^T(\boldsymbol{w})\boldsymbol{e}_i < \boldsymbol{\pi}^T(\boldsymbol{w})\boldsymbol{e}_j. \tag{3.40}$$

This definition captures the intuitive notion of competitiveness: depending on the choice of personalization vector one node is more important than the other one, or viceversa. And, while checking these inequalities in general seems rather complicated, Theorem 3.27 can be shown to provide a powerful way to do so, via the following result [78].

Theorem 3.29. *Given a graph $G = (V, E)$, a fixed damping factor $\alpha \in (0, 1)$, two nodes $i, j \in V$ are effective competitors if and only if there exist $k, \ell \in \{1, \dots, n\}$ such that*

$$x_{ki} > x_{kj} \quad \text{and} \quad x_{\ell i} > x_{\ell j}, \tag{3.41}$$

where $X = (x_{ij}) = (1 - \alpha)(\mathbb{I}_N - \alpha P)^{-1} \in \mathbb{R}^{N \times N}$.

In other words, it is enough to compare the ith-column and the jth-column of matrix X; if each entry of the ith-column is always greater or equal than the corresponding entry of the jth-column then nodes i and j are not effective competitors.

To finalize this recap of the results shown in [78], there the authors define yet another property of nodes in relation to their PageRank scores depending on the personalization vector: leadership.

Definition 3.30 (Leadership group). *Given a graph $G = (V, E)$, we say that $i \in V$ is a leader of G if there exists a personalization vector $\boldsymbol{v} \in \mathbb{R}^N (\boldsymbol{v} > 0, ||\boldsymbol{v}||_1 = 1)$ such that for every node $j \in V$ $(j \neq i)$*

$$\boldsymbol{\pi}^T(\boldsymbol{v})\mathbf{e}_i > \boldsymbol{\pi}^T(\boldsymbol{v})\mathbf{e}_j. \tag{3.42}$$

The set of all leader nodes of a graph G is called the leadership group of the network.

This, again, enables a precise definition of leadership: possible leaders are nodes ranked the highest for some choice of the personalization vector. And, as in the competitors case, there is a result tying Theorem 3.27 to this definition, which enables a simple computation of the leadership group for a given graph [78].

Theorem 3.31. *Given a graph $G = (V, E)$, a fixed damping factor $\alpha \in (0, 1)$, the leadership group of G is the set of nodes $i \in V$ verifying that there is $j \in V$ such that for every $k \in V$ $(k \neq i)$*

$$x_{ji} > x_{jk}, \tag{3.43}$$

where $X = (x_{ij}) = (1 - \alpha)(\mathbb{I}_N - \alpha P)^{-1} \in \mathbb{R}^{N \times N}$.

It is worth mentioning that the localization of PageRank values has recently been generalized to the case of temporal networks in [89].

3.4.2 PageRank rankings and the personalization vector

While the previous results are focused at individual centrality scores, a natural step forward is the analysis of the *entire* centrality outcomes as functions of the parameters. This is something we tackled in [67], which we now proceed to discuss.

In order to answer this problem we derived bounds relating the damping factor and the personalization vector for the complete control problem. They are rather strict, which is why we then relax the problem to that of ranking control, where we can again obtain some bounds which are softer but still strict.

3.4.2.1 Complete control

It is clear that suitable adjustments of the damping factor α and the personalization vector $\boldsymbol{v}$ will be needed in order to fix the PageRank centrality of the given network G (see, for example [65, 78]). What we attempt to do is quantifying the balance between the adjustment of both parameters. In other words, we want to understand what ranges of α provide the desired centrality vector for suitable $\boldsymbol{v}$.

Operating with (2.28) it is straightforward to obtain the following formula [65]

$$\boldsymbol{\pi}^T(\mathbb{I}_N - \alpha P) = (1-\alpha)\boldsymbol{v}^T. \tag{3.44}$$

Traditionally, this equation can be viewed as an equation for $\boldsymbol{\pi}$ given α, $\boldsymbol{v}$ and P. However, we can also view it as an equation for $\boldsymbol{v}$ given α, $\boldsymbol{\pi}$ and P:

$$\boldsymbol{v}^T = \frac{1}{1-\alpha}\boldsymbol{\pi}^T(\mathbb{I}_N - \alpha P). \tag{3.45}$$

This equation tells us which personalization vector is required to obtain a desired PageRank vector, for a fixed network and damping factor. This raises a question: can we always find such non-negative personalization vector that gets a prescribed PageRank centrality? This natural question is summarized in the following problem:

Problem 3.32 (Complete control via the personalization vector). *Given a graph G, a damping factor $\alpha \in (0,1)$ and a positive, unit norm vector $\boldsymbol{\pi}_0$, does it always exists a positive, unit norm $\boldsymbol{v}$ such that the $\boldsymbol{\pi}_0$ is the PageRank outcome?*

In other words: can any PageRank vector be set for a given graph and damping factor if we have control over the personalization vector used in the algorithm?

The answer is no, since there is no positive ($v_i > 0$, $\forall i$) solution in some cases. Nevertheless, we can study the conditions under which $\boldsymbol{\pi}_0$ actually has an associated personalization vector $\boldsymbol{v}$ and the following result give a characterization of the existence of positive personalization vectors that give a prescribed PageRank centrality $\boldsymbol{\pi}_0$ in terms of the size of its components.

Theorem 3.33 (Existence of the personalization vector). *Given a graph G and a positive, unit norm vector $\boldsymbol{\pi}_0$ then there exists a positive, unit norm personalization vector $\boldsymbol{v}$ such that $\boldsymbol{\pi}_0$ is the PageRank vector if and only if $\boldsymbol{\pi}_0^T\boldsymbol{e}_j > \alpha\boldsymbol{\pi}_0^T P\boldsymbol{e}_j$ for all j.*

Proof. First we prove that (3.45) leads to unit norm personalization vectors, since

$$\begin{aligned} ||\boldsymbol{v}||_1 = \boldsymbol{v}^T\boldsymbol{e} &= \frac{1}{1-\alpha}\boldsymbol{\pi}_0^T(\mathbb{I}_N - \alpha P)\boldsymbol{e} = \frac{1}{1-\alpha}\boldsymbol{\pi}_0^T(\boldsymbol{e} - \alpha P\boldsymbol{e}) \\ &= \boldsymbol{\pi}_0^T\boldsymbol{e} = ||\boldsymbol{\pi}_0||_1 = 1, \end{aligned} \tag{3.46}$$

where we used the row-stochasticity in $P\boldsymbol{e} = \boldsymbol{e}$. We now require that all of $\boldsymbol{v}$'s components are positive, so

$$v_j = \boldsymbol{v}\boldsymbol{e}_j = \frac{1}{1-\alpha}\boldsymbol{\pi}_0^T(\mathbb{I}_N - \alpha P)\boldsymbol{e}_j > 0, \tag{3.47}$$

which completes the proof. □

It is also remarkable to point out that Theorem 3.33 presents some analytical interplay between the damping factor and personalization vectors, since if we take a positive, unit norm $\boldsymbol{\pi}_0$ and $0 < \alpha \leq \min_j(\boldsymbol{\pi}_0^T \boldsymbol{e}_j)$ then it can be checked that for any graph without dangling nodes there exists a positive, unit norm personalization vector $\boldsymbol{v}$ such that $\boldsymbol{\pi}_0$ is the PageRank vector. In fact, if we consider a graph without dangling nodes, note that $P\boldsymbol{e}_j$ is the j-th column of P, that is

$$P\boldsymbol{e_j} = \left(\frac{a_{1j}}{k_1^{\text{out}}}, \frac{a_{2j}}{k_2^{\text{out}}}, ..., \frac{a_{Nj}}{k_N^{\text{out}}}\right)^T, \tag{3.48}$$

so we have that $\boldsymbol{\pi}_0^T P\boldsymbol{e}_j \leq \boldsymbol{\pi}_0^T \boldsymbol{e} = 1$, since $0 \leq a_{ij}/k_i^{\text{out}} \leq 1$ and hence, if we take $\alpha < \min_j \boldsymbol{\pi}_0^T e_j$, then

$$\alpha \boldsymbol{\pi}_0^T P e_j \leq \alpha < \boldsymbol{\pi}_0^T e_j \quad \forall 1 \leq j \leq N, \tag{3.49}$$

hence there exists a personalization vector $\boldsymbol{v}$ such that $\boldsymbol{\pi}_0$ is the PageRank vector, simply by using Theorem 3.33.

3.4.2.2 The ranking control problem

The subject of ranking control has remained fairly unexplored due to its technical complexity (as lifting the constraint of fixing concrete centrality vectors makes the problem harder to tackle), but in the PageRank case Theorem 3.33 provides us with a valuable tool to investigate in this direction by using some techniques from convex geometry.

Consider the following milder version of Problem 3.32, where we are now only interested in rankings rather than concrete PageRank vectors.

Problem 3.34 (Ranking control via personalization vector). *Given a graph G, a damping factor $\alpha \in (0, 1)$ and an ordering of the nodes (allowing for ties), does it always exists a positive, unit norm personalization vector $\boldsymbol{v}$ such that the PageRank outcome follows the prescribed order?*

In order to study this problem we will now change the viewpoint of the discussion to a geometric one: consider the N-simplex defined as

$$\Delta_N = \left\{\boldsymbol{x} \in \mathbb{R}^N \mid \boldsymbol{x} > 0, \, ||\boldsymbol{x}||_1 = 1\right\}. \tag{3.50}$$

This set represents the convex span of vectors $\{e_1, ..., e_N\}$, and thus is the space of all possible personalization vectors and the space of all possible PageRank vectors of graphs with n nodes. Therefore, we can understand equation (3.44) as the following map from Δ_N to itself:

$$\begin{aligned} \boldsymbol{\pi}(G, \alpha, \cdot): \quad & \Delta_N \longrightarrow \Delta_N \\ & \boldsymbol{v} \longmapsto \boldsymbol{\pi}(G, \alpha, \boldsymbol{v}). \end{aligned} \tag{3.51}$$

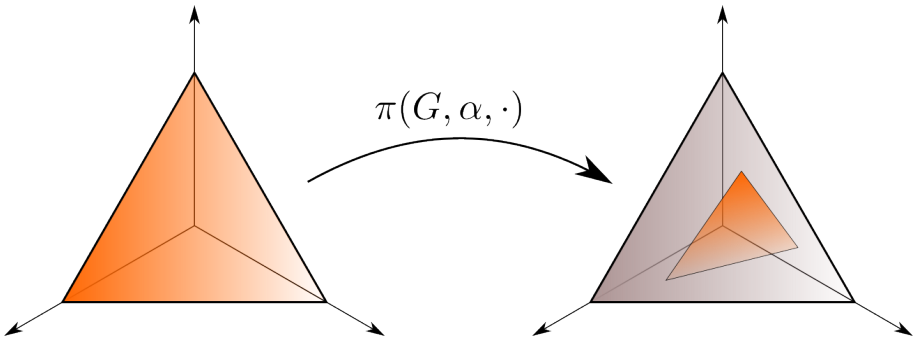


Figure 3.6: Depiction of the map $\boldsymbol{\pi}(G, \alpha, \cdot)$ for $N = 3$.

This map is injective (however in general it is not surjective) and linear in $\boldsymbol{v}$, so $\boldsymbol{\pi}(G, \alpha, \Delta_N)$ is a polytope (i.e. the convex hull of a finite number of points) in $\Delta_N \subset \mathbb{R}^N$. Figure 3.6 illustrates this geometrical interpretation of $\boldsymbol{\pi}(G, \alpha, \cdot)$ in case $n = 3$.

The key point in this geometric viewpoint is that we can associate each possible ranking to a portion of the simplex. If we consider the center point (barycenter) of the simplex Δ_N, given by the normalization of $\boldsymbol{e}$, i.e. $\boldsymbol{e}_0 \equiv \boldsymbol{e}/n = \sum_{i=1}^{N} \boldsymbol{e}_i/n$, then we can define the hyperplanes bisecting the simplex through the center $\boldsymbol{e}_0$ and any combination of $n-2$ vertices as

$$\mathcal{H}_n^{i,j} = \left\{ \sum_{\substack{k=0 \\ k \neq i,j}}^{N} \lambda_k \boldsymbol{e}_k \quad \text{such that} \quad \lambda_k \in \mathbb{R} \right\} \subseteq \mathbb{R}^N. \tag{3.52}$$

The relevance of this construction is that it provides us with a way to classify the points $\boldsymbol{\pi} \in \Delta_N$ according to their ranking. To see this, consider for instance the hyperplane $\mathcal{H}_4^{1,2}$. It can be identified as the region of ranking space where $c_1 = c_2$, by definition. If we move away from it in the direction of $\boldsymbol{e}_2$ we will have $c_1 < c_2$, and viceversa.

In general, the $\binom{N}{2}$ planes $\mathcal{H}_N^{i,j}$ uniquely determine the pairwise inequalities between components i, j of the PageRank vector. The original simplex Δ_N is then divided into $N!$ regions (the number of permutations of the components of the PageRank vector), each of them determining a different ranking. A depiction of these regions for the $N = 3$ case can be seen in Figure 3.7.

In this light, we can see that there is ranking control if and only if

$$\boldsymbol{e}_0 = \frac{1}{N}\boldsymbol{e} \in \mathrm{Im}(\boldsymbol{\pi}) \quad \text{and} \quad \boldsymbol{e}_0 = \frac{1}{N}\boldsymbol{e} \notin \partial \mathrm{Im}(\boldsymbol{\pi}). \tag{3.53}$$

The argument here is identical to that of the hyperplanes: $\boldsymbol{\pi} = \boldsymbol{e}_0$ is the point in ranking space where $c_1 = c_2 = ... = c_N$. Given that all hyperplanes $\mathcal{H}_N^{i,j}$ pass through $\boldsymbol{e}_0$ by construction, all ranking regions are $\varepsilon > 0$ away from it. Thus, moving $\varepsilon > 0$ away in any direction will lead to different rankings.

This idea may be easier to visualize if we take into account Figure 3.6. Notice that in that case the resulting triangle (right) contains points associated to any

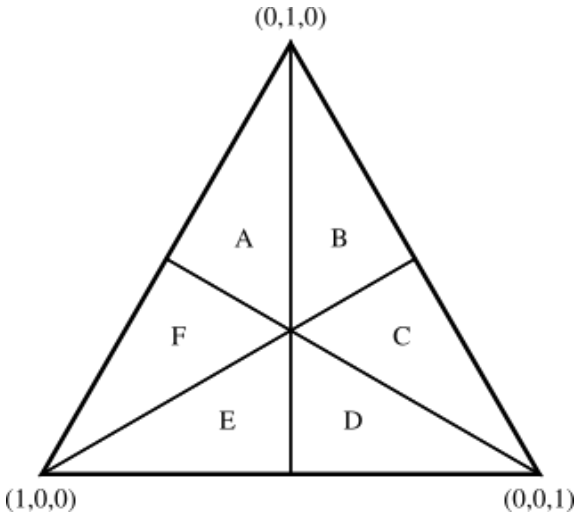


Figure 3.7: Different ranking regions in the $n = 3$ case. For instance, if ∂A denotes the (topological) boundary of $A \subseteq \Delta_N$ in the containing plane of A, then $\boldsymbol{\pi} = (\pi_1, \pi_2, \pi_3) \in A \setminus \partial A$ corresponds to $\pi_2 > \pi_1 > \pi_3$, while the intersection between triangles would lead to equal scores, e.g. $\pi \in B \cap C$ would correspond to $\pi_2 = \pi_3 > \pi_1$.

ranking (as shown in Figure 3.7). The condition necessary and sufficient for this to happen is for it to contain the centerpoint of the bigger triangle.

Next, we can give an analytical characterization of the existence of of a prescribed ranking of the nodes in terms of the relationship between the damping factor and the column sums of P, which is the analogous of Theorem 3.33 but for the Ranking problem.

Theorem 3.35 (Characterization of ranking control). *Given a graph G and damping factor $\alpha = (0, 1)$, then it is possible to obtain any ranking of the nodes under the PageRank if and only if*

$$\frac{1}{\alpha} > \max_j \left(\sum_{i=1}^{N} P_{ij} \right). \tag{3.54}$$

Proof. Using $\boldsymbol{\pi}_0 = \boldsymbol{e}_0$ in Theorem 3.33 yields

$$\boldsymbol{e}_0^T \boldsymbol{e}_j = \frac{1}{N} > \alpha \boldsymbol{e}_0^T P \boldsymbol{e}_j = \frac{1}{N} \alpha \sum_{i=1}^{N} P_{ij}, \tag{3.55}$$

for all $1 \leq j \leq N$, which already gives us the characterization of the existence of a personalization vector that gives any prescribed ranking of nodes. By virtue of the aforementioned Theorem, we can also conclude it to be a sufficient condition for the existence of a personalization vector allowing for any desired ranking. □

Given that $\sum_i P_{ij}$ is the total probability that a random walker visits node j,

this theorem can be interpreted as an upper bound for α in terms of the maximum of these total probabilities. This upper bound tells us that, provided we have $\alpha < 1/\max_j \sum_i P_{ij}$, we can always find any desired ranking with an appropriate choice of personalization vector. It is important to note that this is not a statistical result, in the sense that as long as there is one node targeted by many others with low out-degree, there will be almost no room for ranking control, regardless of the topology of the rest of the network. As we will see later, this is very reminiscent of the scale-free [12] network paradigm: indeed, scale-free networks present these high in-degree nodes pointed to by low out-degree ones.

It is also remarkable to point out the fact that if we denote

$$\alpha_0 = \frac{1}{\max_j \sum_i P_{ij}}, \tag{3.56}$$

then $\alpha_0 \in (0, 1]$ is a measure of the *controllability* of the PageRank in graph G, since the bigger α_0 is the wider range of damping factors allow ranking control of PageRank in G.

3.4.2.3 Real network datasets

Having found an network-specific upper bound for the value of the damping factor α, which would allow the PageRank of the network to be ranking-controllable tinkering with the personalization vector, it is left for us to find out whether it is a hard or soft bound.

The standard value considered for the damping is $\alpha = 0.85$ [2], whose interpretation in terms of Internet hyperlink networks is that of an surfer clicking on hyperlinks ∼8 times before losing interest and searching for something else; this value corresponds to constraining the maximum of the column sum of P to around 1.17. This is clearly a very strict condition.

In fact, we have computed the maximum of the column sums of P for a variety of networks[2], publicly available from different Internet sources (all fetch from the KONECT network repository [48] and the CASOS network repository [46]). We can extract the maximum value of the damping factor α which would enable us to have ranking control over each network's PageRank rankings. This is shown in Figure 3.8.

As expected from the above discussion, the maximum values of the damping factor are generally small compared to the standard $\alpha = 0.85$, regardless of the network size. There are a couple of exceptionally high values, but still lower than such value. We see, on the other hand, that the smaller the network the more controllable it is. This can also be understood from Theorem 3.35: a higher number of nodes means that the maximum column sum of P is likely to be higher (specially due to the number of edges growing also linearly with the number of nodes), hindering controllability.

[2]It should be noted that in order to perform this computation we had to deal with the issue of the dangling nodes, which we glossed over at the beginning of this section. We deemed adding a single, random connection from each dangling node to another, non-danging node as the simplest, least intrusive way to remove this issue.

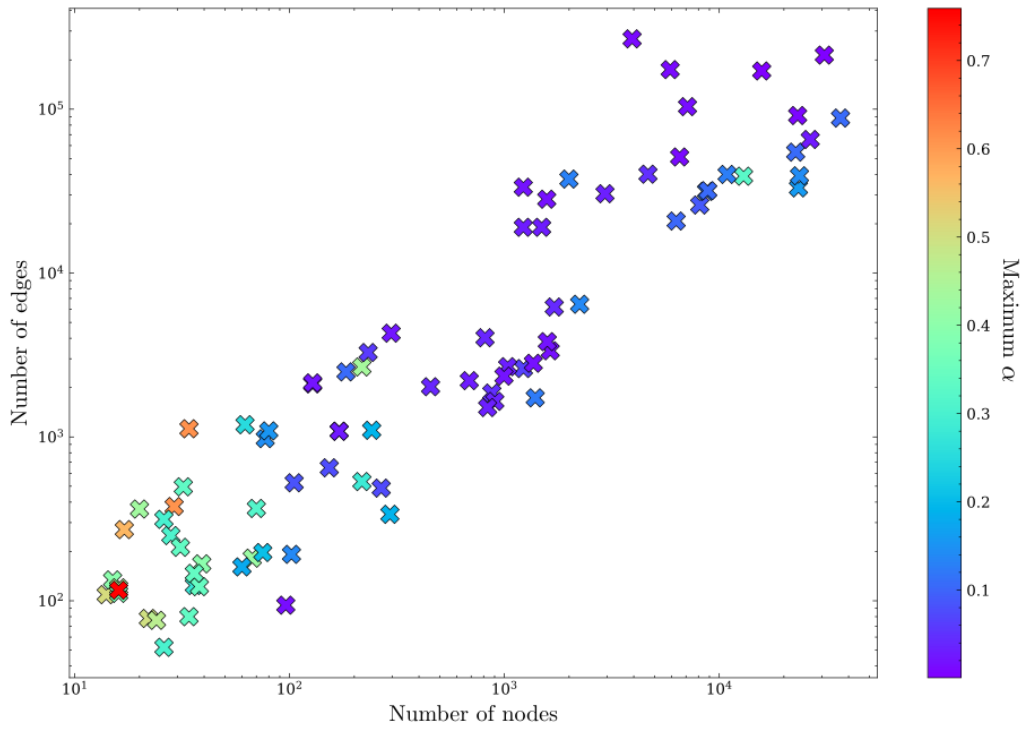


Figure 3.8: Scatter plot showing the number of edges against the number of nodes for 84 different real networks obtained from the KONECT network repository [48] and the CASOS network repository [46], with datapoints colored based on the maximum value of α providing ranking control.

While our results are of a theoretical nature, and thus are not related to any specific implementation or application of PageRank, it might also be interesting to address the implications of this bound in some specific use cases of PageRank (and more concretely, understanding the teleportation vector in them).

- World Wide Web and similar data: here the purpose of PageRank is mainly identifying websites of interest for a given user. The teleportation therefore allows for tweaking the preferences of the user, providing different rankings to a user with a different personalization vector. The bound (3.54) in this case tells us that the ordering is robust: the ranking can't be completely altered by the choice of personalization.

- Genetical or Protein-Protein Interaction networks: PageRank has been applied in a variety of biological networks [90, 91]. In these applications, the teleportation vector is designed to focus the search on specific areas of the network. Given that, as discussed in the aforementioned paper, the damping used in these applications is high ($\alpha \approx 0.8$), we can also to conclude that the ranking will also be robust with respect to changes in the personalization vector.

- Knowledge information systems: PageRank has found a way to be used in semi-supervised learning tasks (for instance, in graphs where each node is an

image and two are connected if they share a caption label [92]). These studies present a different phenomenology to the previously discussed cases, as they employ very low values of $\alpha \approx 0.1$. Although we have no datasets of this type, our results force us to conclude that it is quite likely that the rankings obtained will be highly dependent on the personalization vector used.

- There are plenty other applications (see [22]), most of them using high α. We can draw similar conclusions to the previous cases, the robustness of the ranking.

Some of the datasets used in Figure 3.8 fall into these or other categories. The specific data used in each of them can be found in our GitHub repository (which can be found in the Data Availability section of [67]), and the information about each dataset is in [48, 46].

3.4.3 Beyond PageRank: controlling related measures

As we briefly touched upon in Section 2.3, there are several variants of PageRank which have been proposed over the years. It would be interesting to understand if the analysis carried out in the previous subsections can be extended to them, and if so, what changes.

Here we will briefly examine some of those variants, to find analogous bounds which can then be used to judge how controllable they are, in the same spirit as in the previous Subsections.

3.4.3.1 Ranking control of Biplex PageRank

We start with the biplex PageRank centrality measure, which was introduced at the end of Section 2.3. One can easily notice that the ingredients we used to derive the result of the previous subsection (Theorems 3.33, 3.35, etc) are also present in this measure: there is a damping factor β involved, as well as a personalization vector $\boldsymbol{v}$, whose relation we can disentangle.

Let us first note that most of what was discussed in Subsection 3.4.1 was already extended to this variation of PageRank in [93], we will omit its discussion here for conciseness sake, and focus on the extension of Subsection 3.4.2 to this setting, something we also carried out in [67].

Starting from Definition 2.29, notice that vectors $\boldsymbol{\pi}_u$ and $\boldsymbol{\pi}_d$ satisfy the following relations, $\boldsymbol{\pi}_u \mathbf{e} = \beta$ and $\boldsymbol{\pi}_d \mathbf{e} = 1 - \beta$. Later in [93] a closed form formula for the Biplex PageRank vector, in resemblance to formula (3.44), was found as

$$\boldsymbol{\pi}_{\mathrm{BPR}}^T = (1-\beta)^2 \boldsymbol{v}^T \left(\beta \mathbb{I}_N + Y\right) Z^{-1}, \tag{3.57}$$

where $Y = \mathbb{I}_N - \beta P$, $Z = \gamma \mathbb{I}_N - \beta P$ and $\gamma = 1 - \beta(1-\beta)$. It is straightforward to check that $(\beta \mathbb{I}_N + Y)$ is invertible in the $\beta \in (0,1)$ range, so we also have the formula

$$\boldsymbol{v}^T = \frac{1}{(1-\beta)^2} \boldsymbol{\pi}_{\mathrm{BPR}}^T Z \left(\beta \mathbb{I}_N + Y\right)^{-1}. \tag{3.58}$$

With this, we can state the following theorem that characterizes when a personalization vector exists for a prescribed biplex PageRank centrality.

Theorem 3.36 (Existence of the personalization vector (biplex)). *Given a graph G and a positive, unit norm $\boldsymbol{\pi}_{BPR}$, then there exists a positive, unit norm personalization vector $\boldsymbol{v}$ such that $\boldsymbol{\pi}_0$ is the biplex PageRank vector if and only if $\boldsymbol{\pi}_{BPR}^T \boldsymbol{e}_j > \beta \boldsymbol{\pi}_{BPR}^T \mathcal{P} \boldsymbol{e}_j$ for all j, where $\mathcal{P} = (2-\beta)(\beta \mathbb{I}_N + Y)^{-1}$.*

Proof. First we prove that (3.58) leads to unit-norm personalization vectors. Note that $P^N \boldsymbol{e} = \boldsymbol{e}$ due to row-stochasticity, therefore if we use the resolvent expansion

$$(\beta \mathbb{I}_N + Y)^{-1} = \frac{1}{\beta+1} \sum_{m=0}^{\infty} \left(\frac{\beta}{1+\beta} P \right)^m, \tag{3.59}$$

we end up with

$$\begin{aligned} ||\boldsymbol{v}||_1 = \boldsymbol{v}^T \boldsymbol{e} &= \frac{1}{(1-\beta)^2} \boldsymbol{\pi}_{\mathrm{BPR}}^T Z \left(\beta \mathbb{I}_N + Y\right)^{-1} \boldsymbol{e} \\ &= \frac{1}{(1-\beta)^2} \boldsymbol{\pi}_{\mathrm{BPR}}^T Z \frac{1}{\beta+1} \sum_{m=0}^{\infty} \left(\frac{\beta}{1+\beta} P \right)^m e \\ &= \frac{1}{(1-\beta)^2} \boldsymbol{\pi}_{\mathrm{BPR}}^T (\gamma \mathbb{I}_N - \beta P) \boldsymbol{e} \frac{1}{\beta+1} \sum_{m=0}^{\infty} \left(\frac{\beta}{1+\beta} \right)^m \\ &= \frac{1}{(1-\beta)^2} \boldsymbol{\pi}_{\mathrm{BPR}}^T (1-\beta)^2 \boldsymbol{e} = \boldsymbol{\pi}_{\mathrm{BPR}}^T \boldsymbol{e} = 1. \end{aligned} \tag{3.60}$$

We now require that all components of the required personalization vector are positive,

$$v_j = \boldsymbol{v}^T \boldsymbol{e}_j = \frac{1}{(1-\beta)^2} \boldsymbol{\pi}_{\mathrm{BPR}}^T Z (\beta \mathbb{I}_N + Y)^{-1} \boldsymbol{e}_j > 0. \tag{3.61}$$

It will now be convenient expanding the $Z(\beta \mathbb{I}_N + Y)^{-1}$ expression in with the previously mentioned resolvent series, multiplying and re-summing. Doing so we find

$$\begin{aligned} Z(\beta \mathbb{I}_N + Y)^{-1} &= \frac{1}{\beta+1} (\gamma \mathbb{I}_N - \beta P) \sum_{m=0} \left(\frac{\beta}{\beta+1} \right)^m P^m \\ &= \frac{1}{\beta+1} \sum_{m=0} \left[\gamma \left(\frac{\beta}{\beta+1} \right)^m P^m - \beta \left(\frac{\beta}{\beta+1} \right)^m P^{m+1} \right] \\ &= \frac{1}{\beta+1} \left[\gamma \mathbb{I}_N - \beta(\beta-2) \mathbb{I}_N + \beta(\beta-2) \sum_{m=0}^{\infty} \left(\frac{\beta}{\beta+1} \right)^m P^m \right] \\ &= \mathbb{I}_N + \beta(\beta-2)(\beta+Y)^{-1}. \end{aligned} \tag{3.62}$$

Plugging this in the above equation we find the condition

$$\left[\boldsymbol{\pi}_{\mathrm{BPR}} + \beta(\beta-2) \boldsymbol{\pi}_{\mathrm{BPR}} (\beta \mathbb{I}_N + Y)^{-1} \right] \boldsymbol{e}_j > 0, \tag{3.63}$$

which, with the identification $\mathcal{P} = (2-\beta)(\beta\mathbb{I}_N + Y)^{-1}$ concludes the proof. □

Again, by using the geometric approach proposed in Section 3.4.2.2 we can interpret the biplex PageRank vector as the linear map between simplices (3.50)

$$\begin{aligned} \boldsymbol{\pi}_{\mathrm{BPR}}(G,\beta,\cdot): \quad & \Delta_N \longrightarrow \Delta_N \\ & \boldsymbol{v} \longmapsto \boldsymbol{\pi}_{\mathrm{BPR}}(G,\beta,\boldsymbol{v}). \end{aligned} \tag{3.64}$$

This map is again injective and linear in $\boldsymbol{v}$, and consequently allows us to employ the same kind of argument for the existence of ranking controllability

$$\boldsymbol{e}_0 = \frac{1}{n}\boldsymbol{e} \in \mathrm{Im}(\boldsymbol{\pi}_{\mathrm{BPR}}), \quad \boldsymbol{e}_0 = \frac{1}{n}\boldsymbol{e} \notin \partial\mathrm{Im}(\boldsymbol{\pi}_{\mathrm{BPR}}). \tag{3.65}$$

It is straightforward to find an analytic characterization of ranking control in the biplex PageRank case in terms of the relationship between β and the column sums of matrix P, simply by following the same reasoning used in the standard PageRank setting. In fact, following similar arguments that those used in the proof of Theorem 3.35 it can be easily proved the following result:

Theorem 3.37 (Characterization of biplex ranking control). *Given a graph G and a damping factor $\beta = (0,1)$, then it is possible to obtain any ranking of the nodes under the biplex PageRank if and only if*

$$\frac{1}{\beta} > \max_j \left(\sum_{i=1}^{N} \mathcal{P}_{ij} \right). \tag{3.66}$$

By using the definition of $\mathcal{P}$, the condition that appears in Theorem 3.37 can be rewritten as follows:

$$\frac{1}{\beta} > \max_j \left(\sum_{i=1}^{N} \mathcal{P}_{ij} \right) = \left(\frac{2-\beta}{1+\beta} \right) \max_j \sum_{i=1}^{N} \left[\left(\mathbb{I}_N - \frac{\beta}{1+\beta} P \right)^{-1} \right]_{ij}, \tag{3.67}$$

but we cannot expect a more simplified expression of the maximal β in terms of P_{ij}, since matrix $\mathcal{P}$ depends itself on the damping factor, unlike what happened in the standard PageRank case.

3.4.3.2 Interlude: realization of node-dependent restart PageRank via standard PageRank

We are now interested in analyzing the possibilities of controlling the node-dependent restart PageRank, which was introduced also at the end of Section 2.3. The measure is again equipped with similar ingredients as the standard PageRank, as there is not only one, but a collection of dampings $\alpha_i \in (0,1)$ involved, as well as a personalization vector $\boldsymbol{w} \in \mathbb{R}^N$.

Before analyzing the relation between these and the centralities, it is worth understanding how are they tied when compared to PageRank. In particular, we want to quantify how redundant is this measure as opposed to the vanilla PageRank, in the sense of achieving the same outcomes. This is relevant for control purposes, as we want to know how much more versatility is provided by this generalization.

Theorem 3.38. *Let $G = (V, E)$ be a graph, $\alpha_i \in (0, 1)$, $i \in V$ node-dependent dampings and $\boldsymbol{w} \in \mathbb{R}^N$, $\boldsymbol{w} > 0$, $||\boldsymbol{w}||_1 = 1$ a personalization vector in the node-dependent restart PageRank $\boldsymbol{\pi}_{\text{NPR}}$. There always exists a personalization vector $\boldsymbol{v} \in \mathbb{R}^N$, $\boldsymbol{v} > 0$, $||\boldsymbol{v}||_1 = 1$ such that the standard PageRank coincides with the node-dependent restart PageRank $\boldsymbol{\pi}_{\text{PR}} = \boldsymbol{\pi}_{\text{NPR}}$ if*

$$\boldsymbol{w}_{\mathcal{A}}^T \mathbf{e}_i > \alpha \boldsymbol{w}_{\mathcal{A}}^T P \mathbf{e}_i \quad \forall i \in V, \tag{3.68}$$

where $\boldsymbol{w}_{\mathcal{A}}^T = \boldsymbol{w}^T (\mathbb{I}_N - \mathcal{A}P)^{-1}$.

Proof. Note that from (2.31) one can derive the following equation for the personalization vector $\boldsymbol{w}$ [69]:

$$\boldsymbol{\pi}_{\text{NPR}}(\mathcal{A}, \boldsymbol{w}) = \frac{1}{\gamma} \boldsymbol{w}^T (\mathbb{I}_N - \mathcal{A}P)^{-1}, \tag{3.69}$$

where

$$\gamma = \boldsymbol{w}^T (\mathbb{I}_N - \mathcal{A})^{-1} \mathbf{e} = \sum_{i=1}^{N} \frac{\boldsymbol{w}_i}{1 - \alpha_i} > 1. \tag{3.70}$$

The condition we therefore must check is

$$\boldsymbol{\pi}_{\text{NPR}}(\mathcal{A}, \boldsymbol{w}) = \boldsymbol{\pi}_{\text{STD}}(\alpha, \boldsymbol{v}) \Rightarrow \frac{1}{\gamma} \boldsymbol{w}^T (\mathbb{I}_N - \mathcal{A}P)^{-1} = (1 - \alpha) \boldsymbol{v}^T (\mathbb{I}_N - \alpha P)^{-1}. \tag{3.71}$$

We can then write v^T explicitly as

$$\boldsymbol{v}^T = \frac{1}{\gamma(1 - \alpha)} \boldsymbol{w}^T (\mathbb{I}_N - \mathcal{A}P)^{-1} (\mathbb{I}_N - \alpha P). \tag{3.72}$$

We need to prove that this expression is positive and consistent with unit-normalization. Starting with the latter,

$$\boldsymbol{v}^T \mathbf{e} = \frac{1}{\gamma(1 - \alpha)} \boldsymbol{w}^T (\mathbb{I}_N - \mathcal{A}P)^{-1} (1 - \alpha) \mathbf{e} = \frac{1}{\gamma} \boldsymbol{w}^T (\mathbb{I}_N - \mathcal{A}P)^{-1} \mathbf{e} = 1. \tag{3.73}$$

As for positivity, we have

$$v_i = \boldsymbol{v}^T \mathbf{e}_i = \frac{1}{\gamma(1 - \alpha)} \boldsymbol{w}^T (\mathbb{I}_N - \mathcal{A}P)^{-1} (\mathbb{I}_N - \alpha P) \mathbf{e}_i > 0, \quad \forall i. \tag{3.74}$$

Notice that $1 - \alpha > 0$ and $\gamma \geq 1$, therefore this condition can be written as

$$\boldsymbol{w}^T (\mathbb{I}_N - \mathcal{A}P)^{-1} \mathbf{e}_i > \alpha \boldsymbol{w}^T (\mathbb{I}_N - \mathcal{A}P)^{-1} P \mathbf{e}_i, \quad \forall i. \tag{3.75}$$

Defining $\boldsymbol{w}_{\mathcal{A}}^T = \boldsymbol{w}^T(\mathbb{I}_N - \mathcal{A}P)^{-1}$, we have that $\boldsymbol{w}_{\mathcal{A}}^T > 0$, and

$$\boldsymbol{w}_{\mathcal{A}}^T \mathbf{e}_i > \alpha \boldsymbol{w}_{\mathcal{A}}^T P \mathbf{e}_i = \alpha \sum_{j=1}^{N} (\boldsymbol{w}_{\mathcal{A}})_j^T P_{ji}, \quad \forall i, \tag{3.76}$$

concluding the argument. □

This result shows that, if we can change both α and $\boldsymbol{v}$, the standard PageRank can always find that value of the node-dependent restart PageRank. In hindsight, this rather sensible: in the limit $\alpha \to 0$, $\boldsymbol{v}$ dictates the centralities.

A more nuanced question is then the reverse one: can we conceive conceive a centrality $\boldsymbol{\pi}_0 \in \mathbb{R}^N$, with $\boldsymbol{\pi}_0 > 0$, $||\boldsymbol{\pi}_0||_1 = 1$ which can't be realized for a given α in the standard PageRank (see Theorem 3.33 for this condition), which nevertheless can be obtained from higher values of the dampings $\alpha_i > \alpha$ in the node-dependent restart case, due to the additional freedom it provides? This is something we will soon discuss.

3.4.3.3 Localization of node-dependent restart PageRank

We can derive similar results as those in [78], discussed in Subsection 3.4.1, for the node-dependent restart PageRank, regarding the localization of each PageRank scores and subsequently of the possible competitors.

Lemma 3.39. *Let P be a row-stochastic matrix, $\alpha_i \in (0,1)$, $\forall i$, $\mathcal{A} = diag(\alpha_1, ..., \alpha_N)$ and $\gamma = [\boldsymbol{w}^T(\mathbb{I}_N - \mathcal{A})^{-1}\mathbf{e}]^{-1}$ with $\boldsymbol{w} \in \mathbb{R}^N$, $\boldsymbol{w} > 0$, $|\boldsymbol{w}|_1 = 1$. Then*

- *$Y = \mathbb{I}_N - \mathcal{A}P$ is strictly row-diagonally dominant.*
- *$X = \gamma\, Y^{-1}$ is strictly diagonally dominant of its column entries.*
- *The i^{th} column of X attains its maximum value at x_{ii}.*

Proof. Firstly, as P is row-stochastic ($P\mathbf{e} = \mathbf{e}$) the sum of each row of Y is $(Y\mathbf{e})_i = (\mathbf{e} - \mathcal{A}P\mathbf{e})_i = (\mathbf{e} - \mathcal{A}\mathbf{e})_i = 1 - \alpha_i$. Therefore, since $\alpha \in (0,1)$ and $0 \leq p_{jk} \leq 1$ for all $i, k = 1, ..., N$ we get

$$|y_{ii}| = |1 - \alpha_i p_{ii}| = 1 - \alpha_i p_{ii} = 1 - \alpha_i + \alpha_i \sum_{k \neq i} p_{ik} > \alpha_i \sum_{k \neq i} p_{ik} = \sum_{k \neq i} |y_{ik}|, \tag{3.77}$$

where we used the fact that P is row-stochastic. We have therefore shown that Y is strictly row-diagonally dominant. By Theorem 2.5.12 of [94] we know that Y^{-1} and X are strictly diagonally dominant of their column entries. Therefore,

$$|x_{ii}| > |x_{ki}| \quad \forall i, k \neq i. \tag{3.78}$$

On the other hand, Y is a non-singular M-matrix (see above, taking $s = 1$, as the spectral radius of $\mathcal{A}P$ is less than 1), thus $Y^{-1} \geq 0$ [33]. Hence, the absolute

values of the above formula can be deleted and we have

$$\max_k x_{ki} = x_{ii}, \tag{3.79}$$

finishing the proof. □

Armed with this lemma, we can tackle the localization problem.

Definition 3.40. *Given a graph $G = (V, E)$ and fixed node-dependent dampings $\alpha_i \in (0, 1)$, $i = 1, ..., N$, for each node $i \in V$ we define $\mathcal{PR}(i)$ as the set of all possible values of the node-dependent restart PageRank of node i, i.e.*

$$\mathcal{PR}(i) = \{\boldsymbol{\pi}^T(\boldsymbol{w})\mathbf{e}_i \quad \textit{for all} \quad \boldsymbol{w} \in \mathbb{R}^N,\ \boldsymbol{w} > 0,\ ||\boldsymbol{w}||_1 = 1\} \tag{3.80}$$

The concrete values of $\mathcal{PR}(i)$ can be quantified with the following theorem:

Theorem 3.41. *Under the same conditions and notation as the above definition,*

$$\mathcal{PR}(i) = (\min_j x_{ji}, x_{ii}). \tag{3.81}$$

Proof. There are two steps involved

1. We first want the upper and lower bounds. Without loss of generality, $i = 1$. Then, $\boldsymbol{\pi}^T(\boldsymbol{w})\mathbf{e}_1 = \sum_{j=1}^N w_j x_{j1}$. Now, as $\boldsymbol{w} > 0$ and $||\boldsymbol{w}||_1 = 1$, we have

$$\min_j x_{ji} < \sum_{j=1}^N v_j x_{j1} < \max_j x_{j1} = x_{ii} \tag{3.82}$$

   where the last equality is due to Lemma 3.39.

2. We now want to see that all values within the interval can be found with suitable $\boldsymbol{w}$. Again without loss of generality $i = 1$. Define the convex combination vector

$$\boldsymbol{w}_\lambda^\epsilon = \lambda \boldsymbol{w}_1^\epsilon + (1 - \lambda)\boldsymbol{w}_{j_1}^\epsilon > 0, \quad \lambda \in (0, 1), \tag{3.83}$$

   where j_1 is the minimum of the first column of X and

$$\begin{aligned} \boldsymbol{w}_1^\epsilon &= \left(1 - \epsilon, \frac{\epsilon}{n-1}, \frac{\epsilon}{n-1}, ..., \frac{\epsilon}{n-1}\right)^T, \\ \boldsymbol{w}_{j_1}^\epsilon &= \left(\frac{\epsilon}{n-1}, ..., \underbrace{1 - \epsilon}_{j_1}, ..., \frac{\epsilon}{n-1}\right)^T. \end{aligned} \tag{3.84}$$

   This vector satisfies

$$\lim_{\lambda \to 1} \lim_{\epsilon \to 0} \boldsymbol{\pi}^T(\boldsymbol{w}_\lambda^\epsilon)\mathbf{e}_1 = x_{11}, \quad \lim_{\lambda \to 0} \lim_{\epsilon \to 0} \boldsymbol{\pi}^T(\boldsymbol{w}_\lambda^\epsilon)\mathbf{e}_1 = x_{j_1 1} \tag{3.85}$$

   Hence, for every x with $x_{j_1 1} < x < x_{11}$ there exists some $\epsilon_0, \lambda_0 \in (0, 1)$ such that $\boldsymbol{\pi}^T(\boldsymbol{w}_{\lambda_0}^{\epsilon_0})\mathbf{e}_1 = x$.

□

3.4.3.4 Ranking control of node-dependent restart PageRank

Here we will pick up the discussion of the possible centralities realized with the node-dependent restart PageRank, as compared to the standard PageRank. We are interested in examining if there is a trade-off between having several, node-dependent dampings (which provides more freedom in the measure's parameters) and the actual values of these dampings.

For instance, one could wonder if lowering substantially the values of certain dampings could compensate for higher ones (which restrict the random walk to follow the network structure more closely), allowing for an enhance in controllability. However, as we will see, this is not the case: in fact, the node-dependent restart PageRank is constrained by the standard one.

We will follow closely the narrative from Subsection 3.4.2. Starting from (2.31) we can derive the following formula for the personalization vector $\boldsymbol{w} \in \mathbb{R}^N$ as a function of the centrality $\boldsymbol{\pi} \in \mathbb{R}^N$

$$\boldsymbol{w}^T = \frac{1}{\boldsymbol{\pi}^T(\mathbb{I}_N - \mathcal{A})\mathbf{e}}\boldsymbol{\pi}^T(\mathbb{I}_N - \mathcal{A}P). \tag{3.86}$$

We can use this formula to derive the following bound, akin to Theorem 3.33.

Theorem 3.42 (Existence of the personalization vector). *Given a graph $G = (V, E)$ and a positive, unit norm vector $\boldsymbol{\pi}_0$, then there exists a positive, unit norm personalization vector $\boldsymbol{w} \in \mathbb{R}^N$ such that the node-dependent restart PageRank is $\pi_{\mathrm{NPR}} = \pi_0$ if and only if $\boldsymbol{\pi}_0^T\mathbf{e}_j > (\max_i \alpha_i)\boldsymbol{\pi}_0^T P\mathbf{e}_j$ for all j.*

Proof. We need to show under which conditions does $\boldsymbol{w}$ have unit norm and positivity, for it to be a personalization vector.

First we check the unit-norm,

$$||\boldsymbol{w}||_1 = \boldsymbol{w}^T\mathbf{e} = \frac{1}{\boldsymbol{\pi}_0^T(\mathbb{I}_N - \mathcal{A})\mathbf{e}}\boldsymbol{\pi}_0^T(\mathbb{I}_N - \mathcal{A}P)\mathbf{e} = ||\boldsymbol{\pi}_0|\dot{\mathrm{t}}| = 1. \tag{3.87}$$

where we used the row-stochasticity of P, i.e. $P\mathbf{e} = \mathbf{e}$. As for positivity,

$$\begin{aligned}\boldsymbol{w}_j = \boldsymbol{w}\mathbf{e}_j &= \frac{1}{\boldsymbol{\pi}_0^T(\mathbb{I}_N - \mathcal{A})\mathbf{e}}\boldsymbol{\pi}_0^T(\mathbb{I}_N - \mathcal{A}P)\mathbf{e}_j \\ &> \frac{1}{\boldsymbol{\pi}_0^T(\mathbb{I}_N - \mathcal{A})\mathbf{e}}\boldsymbol{\pi}_0^T(\mathbb{I}_N - (\max_i \alpha_i)P)\mathbf{e}_j > 0,\end{aligned} \tag{3.88}$$

which completes the proof. □

For the ranking control, things are slightly more inconvenient in this case, compared to the standard PageRank. In the closed-form formula for the PageRank vector (3.44) the normalization is independent of the personalization vector $\boldsymbol{v}$ used, it is just $1 - \alpha$. Here, however, the normalization (3.70) in (3.69) depends on the

choice of personalization vector $\boldsymbol{w}$ under consideration. It is therefore interesting for us to leave the normalization behind, and consider the following map:

$$\begin{aligned} \widetilde{\boldsymbol{\pi}}(\mathcal{A}, \cdot) = \quad & \triangle_N \to CH_N \\ & \boldsymbol{w} \hookrightarrow \boldsymbol{w}^T (\mathbb{I}_N - \mathcal{A}P)^{-1}, \end{aligned} \tag{3.89}$$

where CH_N stands for the positive convex hull in N dimensions

$$CH_N = \left\{ \mathbf{x} \in \mathbb{R}^N \;\middle|\; \mathbf{x} = \sum_{i=1}^{N} a_i \mathbf{e}_i,\ a_i > 0 \right\}. \tag{3.90}$$

This map is linear in $\boldsymbol{w}$, therefore it maps the N-dimensional "normalized" simplex $\triangle_N$ into an N-dimensional simplex (possibly un-normalized) in the convex hull. In this new setting, the generalization of Theorem 3.35 requires the existence of a personalization vector such that $\widetilde{\boldsymbol{\pi}}_0$ is in the parameterized line centered in the convex hull.

Theorem 3.43. *Given a graph G and a set of damping factors $\alpha_i = (0, 1)$, $i \in V$, then it is possible to obtain any ranking of the nodes under the node-dependent restart PageRank if and only if*

$$\frac{1}{\max_k \alpha_k} > \max_j \left(\sum_i^N P_{ij} \right). \tag{3.91}$$

Proof. Firstly, the relation between $\widetilde{\boldsymbol{\pi}}$ and $\boldsymbol{\pi}_{\text{NDR}}$ is simply a proportionality factor, which therefore can't change the relative ranking between the components.

Second, the condition for the existence of ranking control is for the line parameterized by $\tilde{\mathbf{e}} = (t, t, \ldots, t) = t\,\mathbf{e} \in \mathbb{R}^N$ with $t > 0$ to pass through the image of $\widetilde{\boldsymbol{\pi}}$.

Imposing this in Theorem 3.42 we easily arrive to the set of restrictions $1 > \mathbf{e}^T \mathcal{A} P \mathbf{e}_j$ for all j. We can restrict it further taking maximums as

$$1 > \mathbf{e}^T \mathcal{A} P \mathbf{e}_j \geq (\max_k \alpha_k) \left(\sum_i^N P_{ij} \right), \forall j \quad \Rightarrow \quad 1 \geq (\max_k \alpha_k) \max_j \left(\sum_i^N P_{ij} \right) \tag{3.92}$$

thus concluding the proof. □

The punchline is that, actually, the freedom in having a collection of damping factors rather than a global one does not guarantee an improvement in terms of ranking control with respect to the standard PageRank: as a matter of fact, the standard PageRank with the damping factor equal to highest one from the collection (hence, the most "restrictive" one) has the same ranking control inequality.

Chapter 4
Extending spectral centralities in hypergraphs

Up until now we have focused on analyzing centrality measures in standard and multilayer networks. That is a reasonable endeavour, as most of the studies and results (centrality or otherwise) involving networks have, until recently, been devoted to systems endowed with pairwise interactions.

Lately, however, the realization that plenty of systems feature interactions beyond pairwise has led the network science community down the path of adapting and extending all previously known results to this new setting. We already introduced some basic notions of hypergraph theory in Section 2.2.4, and in this chapter we will exploit some of them to redefine spectral centralities (in particular, the eigenvector one) in the hypergraph context. The reader is advised to read the second half of Appendix A, in order to achieve a better understanding of some important tensor algebra results which will be used throughout.

We will begin by reviewing the first attempt to generalize the eigenvector centrality to undirected hypergraphs in Section 4.1. After discussing the shortcomings of this method, we will show how to fix it in Section 4.2. That would complete the formulation of eigenvector-like centralities in undirected hypergraphs, hence the next step is understanding the directed hypergraph case. As we will see in Section 4.3, the concept of directedness in higher order interactions is far from trivial, therefore we will need to make sense of it before discussing centralities. We end this chapter with a brief analysis of the controllability of these new measures in Section 4.5.

4.1 Spectral centralities in hypergraphs

When one sets out to redefine the eigenvector centrality, it is natural to start by wondering whether the original heuristic is still sensible. In that regard, the phrase *not only the number of connections matter, but also how important they are* was written down mathematically in equation (2.21).

Notice that that is the simplest way one can "mathematize" said heuristics, but more generally one can think of the following equation

$$f(c_i) = \alpha \sum_{j=1}^{N} A_{ij}\, g(c_j), \tag{4.1}$$

for some functions $f, g : \mathbb{R} \to \mathbb{R}$. The eigenvector centrality (2.21) is the case $f(c) = g(c) = c$.

In a pairwise network this expression is general enough, but in a hypergraph containing interactions of different orders we are missing contributions from interactions beyond pairwise, e.g. those coming from 3-interactions (described by the 3-tensor $T^{(3)}_{ijk}$), 4-interactions (described by the 4-tensor $T^{(4)}_{ijkl}$), etc. Introducing the information from those terms results in

$$f(c_i) = \alpha_2 \sum_{j=1}^{N} A_{ij}\, g_2(c_j) + \alpha_3 \sum_{j,k=1}^{N} T^{(3)}_{ijk}\, g_3(c_j, c_k) + \alpha_4 \sum_{j,k,l=1}^{N} T^{(4)}_{ijkl}\, g_4(c_j, c_k, c_l) + \dots \quad (4.2)$$

where functions $g_m : \mathbb{R}^{m-1} \to \mathbb{R}$ aggregate the centrality scores of neighbors at each order of interaction.

This expression lacks a general solution for arbitrary functions f, g_m. A first approximation would be focusing on a specific order m we want to examine in detail, and dismissing all interactions from other orders. In that sense, we are only allowing the computation of centralities of m-uniform hypergraphs (see Definition 2.24). This is a rather drastic approach, which we will attempt to mitigate in Section 4.2.

Even within this approximation, there is little we can do without imposing explicit choices for functions f, g_m, based on similarity with the standard eigenvector centrality (2.21) case, as well as with simplicity and the availability of theorems guaranteeing the existence and uniqueness of a positive solution. This choices, in the context of uniform hypergraphs, are discussed in the seminal paper [26], which we now briefly review.

4.1.1 Clique reduction and CEC

Let's consider the simplest non-trivial case, $m = 3$ (the generalization to k-uniform is straightforward).

A first idea would be to consider a linear combination of the centralities of the neighbors in each "coupling function" g_m. It turns out that this leads to the eigenvector equation of something called the motif adjacency matrix $W = (W_{ux})$, whose W_{ix} component is the number of hyperedges containing nodes i and x.

Definition 4.1 (CEC of a hypergraph [26]). *Let $H = (V, E)$ be a 3-uniform hypergraph. Its clique-reduction eigenvector centrality (CEC) is defined as the Perron (unique, positive) eigenvector $\mathbf{c}$ of the motif adjacency matrix W, i.e.*

$$\lambda \mathbf{c} = W^T \mathbf{c} \quad \Rightarrow \quad \lambda c_i = \sum_{(i,j,k)} c_j + c_k = \sum_{e \in E, (i,x) \subset e} W_{ix} c_x, \quad \mathbf{c} > 0,\ \lambda = \rho(W). \quad (4.3)$$

where (i, x) is an ordered pair.

The problem with this centrality is that it does not leverage the nonlinear information which hypergraphs could provide: after all, we are simply computing the eigenvector centrality of a different graph which is the "projection" of the original

hypergraph, in the sense that matrix W codifies the amount of hyperedges containing each pairwise interaction, albeit losing information on which interactions where available in the first place.

We are interested in exploring measures which distill nonlinear relations in the data. For that, we will have to rely on different coupling functions, as we will see. Nevertheless, the study of the CEC brings an important tool to the table: it essentially "projects" down higher order interactions into their pairwise constituents. A generalization of this operation is something which we will find quite useful in the next section, when we try to "uniformize" a hypergraph.

4.1.2 $\mathcal{Z}, \mathcal{H}$-eigenvectors and ZEC, HEC

We now want to study fully-fledged tensorial descriptions of spectral centralities. For that, we assume the definition of adjacency tensor Definition 2.25 already introduced in Chapter 1. We first need to establish an operation which will accompany us throughout.

Definition 4.2 (Tensor apply). *Let $\mathcal{T} = (T_{i_1 \dots i_m}) \in \mathbb{R}^{[m,N]}$ be a tensor, let $\mathbf{c} \in \mathbb{R}^N$ be a vector. The tensor apply [26] (also known as TTSV1 [95]) operation $\mathbb{R}^{[m,N]} \times \mathbb{R}^N \to \mathbb{R}^N$ is defined as*

$$[\mathcal{T}\mathbf{c}^{m-1}]_{i_1} = \sum_{i_2,\dots,i_m}^{N} T_{i_1,\dots,i_m} \prod_{k=2}^{m} c_{i_k}. \tag{4.4}$$

With this operation in mind, let's again consider the simplest non-trivial case, an $m = 3$ uniform hypergraph (as before, the generalization to k-uniform is straightforward), with tensor $\mathcal{T} \in \mathbb{R}^{[3,N]}$. The $\mathcal{Z}$-eigenproblem for this tensor can be stated as

$$\lambda \mathbf{c} = \mathcal{T}\mathbf{c}^2 \quad \Rightarrow \quad \lambda c_i = \sum_{j,k}^{N} T_{ijk}\, c_j c_k, \tag{4.5}$$

where $\mathcal{T}\mathbf{c}^2$ is the previously defined "tensor apply" operation. This equation needs to be supplemented by a normalization constraint, usually $|\mathbf{c}|_1 = 1$ or $|\mathbf{c}|_2 = 1$.

The need for this constraint is the following: consider the eigenpair $(\lambda, \mathbf{c})$. If it is a solution to equation (4.5) alone, then so is $(\alpha\lambda, \alpha\mathbf{c})$ for any scaling $\alpha \in \mathbb{R}$. This is rather inconvenient; eigenvalues should be unique. For this reason, one needs to impose the constraint in the "length" of $\mathbf{c}$ to disable scaling λ.

This constraint is somewhat cumbersome: we are trying to generalize the eigenvector centrality, which allows for scaling eigenvectors without further considerations. This is one of the reasons why the following problem comes out to be more natural.

The $\mathcal{H}$-eigenproblem for that same tensor can be stated as

$$\lambda \mathbf{c}^{[2]} = \mathcal{T}\mathbf{c}^2 \quad \Rightarrow \quad \lambda c_i^2 = \sum_{j,k}^{N} T_{ijk}\, c_j c_k, \tag{4.6}$$

with the same "tensor apply" operation as before, and with the notation $\mathbf{c}^{[m-1]}$ indicating the Hadamard (or componentwise) power of vector $\mathbf{c}$.

Notice that here we do not need a normalization constraint; if $(\lambda, \mathbf{c})$ is a solution eigenpair then so is $(\lambda, \alpha\mathbf{c})$ for any scaling $\alpha \in \mathbb{R}$, just as in the standard eigenvector case. Another way to look at this is the fact that this eigenproblem has the "right" units: if we measure vector $\mathbf{c}$ in some unit, the units on the right hand side coincide with those on the left hand side, unlike in equation (4.5).

Apart from the normalization or lack thereof, what are the similarities and differences between both problems? Regarding the similarities, we are in luck, as we can assert that there exist positive solutions to these eigenproblems in the case of strongly connected hypergraphs (as defined in Subsection 2.2.4) due to respective generalizations of the Perron-Frobenius Theorem [96, 97] (See Appendix A).

Similarities end there, as there are several reasons as to why Z-eigenvectors are far more inconvenient to consider in practical scenarios.

- Even though existence and positivity is indeed guaranteed in both cases when the hypergraph is strongly connected, the same can not be said about uniqueness. In the $\mathcal{H}$-eigenproblem case the generalized Perron-Frobenius theorem [97] does also guarantee uniqueness, but in the $\mathcal{Z}$ case it does not.

  A simple example of this is the following [98].

  Example 4.3. *Consider the following tensor* $\mathcal{T} = (T_{ijkl}) \in \mathbb{R}^{[4,2]}$*:*

  $$\begin{aligned} &T_{1111} = T_{2222} = \tfrac{4}{\sqrt{3}}, \quad T_{1112} = T_{1121} = T_{1211} = T_{2111} = 1, \\ &T_{1222} = T_{2122} = T_{2212} = T_{2212} = T_{2221} = 1, \text{ and } \quad T_{ijkl} = 0 \quad \text{elsewhere.} \end{aligned} \tag{4.7}$$

  This tensor is irreducible (its associated hypergraph is obviously strongly connected as it only has two nodes an several hyperedges over them).

  One can check that both $\mathbf{c}_1 = (\frac{\sqrt{3}}{2}, \frac{1}{2})$ *and* $\mathbf{c}_2 = (\frac{1}{2}, \frac{\sqrt{3}}{2})$ *are positive* $\mathcal{Z}$*-eigenvectors (satisfying the* $||\mathbf{c}||_2 = 1$ *normalization constraint) associated to the spectral radius* $\lambda_1 = \frac{11}{2\sqrt{3}}$*.*

- There exists a power-like method for the numerical computation of leading $\mathcal{H}$-eigenvectors [99] (which are precisely the ones we will be interested in), while no such thing exists for $\mathcal{Z}$-eigenvectors. Furthermore, numerical computation of $\mathcal{Z}$-eigenvectors has some convergence problems [97].

Both eigenproblems can be cast in a network science context as centrality measures [26], bearing in mind that they should, in principle, be applied to the transposed versions of the hypergraph's associated tensor $\mathcal{T}^t$ (whose meaning will be discussed later on), as in the standard eigenvector centrality (2.21).

Definition 4.4 (ZEC of a hypergraph [26]). *Let* H *be the a 3-uniform hypergraph, with an associated tensor* $\mathcal{T} = (T_{ijk}) \in \mathbb{R}^{[3,N]}$*. Its* $\mathcal{Z}$*-eigenvector centrality (ZEC) is*

defined as a (positive, possibly non-unique) $\mathcal{Z}$-eigenvector $\mathbf{c}$ *of* $(\mathcal{T})^t$*, i.e.*

$$\lambda \mathbf{c} = (\mathcal{T})^t \mathbf{c}\mathbf{c} \quad \Rightarrow \quad \lambda c_i = \sum_{j,k} (T_{ijk})^t \, c_j c_k, \quad \mathbf{c} > 0, \, \lambda = \rho((\mathcal{T})^t), \tag{4.8}$$

supplemented by a normalization $||\mathbf{c}||_1 = 1$ *or* $||\mathbf{c}||_2 = 1$.

Definition 4.5 (HEC of a hypergraph [26]). *Let H be the a 3-uniform hypergraph, with an associated tensor $\mathcal{T} = (T_{ijk}) \in \mathbb{R}^{[3,N]}$. Its $\mathcal{H}$-eigenvector centrality (HEC) is defined as the Perron-like (unique, positive) $\mathcal{H}$-eigenvector* $\mathbf{c}$ *of* $(\mathcal{T})^t$*, i.e.*

$$\lambda \mathbf{c}^{[2]} = (\mathcal{T})^t \mathbf{c}\mathbf{c} \quad \Rightarrow \quad \lambda c_i^2 = \sum_{j,k} (T_{ijk})^t \, c_j c_k, \quad \mathbf{c} > 0, \, \lambda = \rho((\mathcal{T})^t). \tag{4.9}$$

Notice that both ZEC and HEC are not exactly the same as the $\mathcal{Z}, \mathcal{H}$-eigenproblems: there is a transposition step involved. In Section 4.3 we will come back to this point, but for the time being we are going to restrict ourselves to undirected hypergraphs, where we can just omit the transposition step.

One last comment before moving on: due to the many drawbacks of $\mathcal{Z}$-eigenvectors (and there are more to come), we will mainly focus on HEC in the coming sections and discussions therein. It will be clear from the context, or noted otherwise, when something is HEC/ZEC specific.

4.1.3 Vector centralities

Even though we will restrict our discussion to the spectral centralities we just described, there are a few other proposals which have been put forward to deal with spectral centralities in hypergraphs, which circumvent explicitly solving equation (4.2). Here we will review one of them [100], which will be used for comparisons later on.

The idea of vectorial centralities has been around for a while, and in fact in the multilayer network case we already discussed it at the end of Subsection 2.3.3. In the hypergraph case one could consider the centrality of node i to be a vector $\mathbf{c}^{(i)} \in \mathbb{R}^M$, where $\mathbf{c}_m^{(i)}$ corresponds to the HEC/ZEC centrality score of the node in the m-uniform hypergraph consisting of all order m hyperedges of the original hypergraph. This has two problems: first, for this to be consistent we would need strong-connectedness at each order in the hypergraph, something quite unrealistic. Second, the interplay between different orders is completely absent, leading to possibly very different centrality scores per order.

An attempt to solve those problems was put forward in [100], where they resort to the line graph of a hypergraph (a structure which is proven to be strongly connected if the original hypergraph is as well) [101] to translate the problem to that of hyperedge centrality scores, which can be tackled using standard, pairwise eigenvector centrality yielding $\mathbf{c}(h) \in \mathbb{R}^+$, $h \in E$. These hyperedge centrality components are then "shared"

among the nodes participating in each of them, at each level k, conforming a vectorial centrality score per node, where each component is associated to each order $2, \dots, M$.

$$\mathbf{c}^i = (c_2^i, c_2^i, \dots, c_M^i) \in \mathbb{R}^{M-1},\ i \in V, \quad c_k^i = \frac{1}{k} \sum_{\substack{i \in h \in E \\ |h|=k}} c(h). \tag{4.10}$$

4.2 Moving past the uniform constraint

We now want to tackle the centrality problem as a whole, from equation (4.2) directly [102]. For that, we will first construct a uniform hypergraph from the original, non-uniform one, in a procedure which we call the "uplift". We then show that the construction, if weights are chosen appropriately, furnishes a centrality measure which is consistent with Definition (4.2) in the HEC case, and we supplement this construction with a projection of higher order edges to provide a family of measures (m-UPHEC) allowing us to discern the importance of each node in any (connected) hypergraph. Several numerical comparisons are then shown which confirm the validity and interest of this measure. We left in Appendix B the discussion of the use of the uplift in $\mathcal{Z}$-eigenproblems, which is not valid for centrality purposes but it can be used for guaranteeing uniqueness and existence properties in certain tensors.

4.2.1 The uplift

We start with the uniformization of the hypergraph from the bottom up. For that, consider a hypergraph $H = (V, E)$ whose maximum hyperedge size is M, and a size $m \geq M$, possibly with multiset hyperedges, i.e. hyperedges where the same node is contained more than once (hypergraphs with such particularity are sometimes referred to as "multihypergraphs" [97]). We can turn every hyperedge of size lower than m into that size by adding an auxiliary node[1], which we name "$\star$", possibly multiple times within the same hyperedge.

More precisely, we have the following definition.

Definition 4.6 (Uplifted hypergraph at order m). *Let $H = (V, E)$ be a hypergraph whose maximum hyperedge size is M and let $m \geq M$. We define the uplifted hypergraph at order m as*

$$\widetilde{H} = (\widetilde{V}, \widetilde{E}), \quad \text{where} \quad \widetilde{V} = V \cup \{\star\} \quad \text{and} \quad \widetilde{E} = \left\{ e \cup \left(\bigcup_{l=0}^{m-|e|} \{\star\} \right), e \in E \right\}. \tag{4.11}$$

Remark 4.7. *Note that, from a set-theoretic point of view, uplifted hyperedges are multiset objects, i.e. they may contain the auxiliary node $\star$ more than once.*

[1] This notion of adding extra nodes is already present in other hypergraph-related works, although with completely different purposes: in [103] they call it "augmentation" and use it for community detection, in [104] they call it "inflation" and use it for hypergraph polynomials. In either case they use a simpler, unweighted version, which only encodes adjacency and not strength of it, in contrast with our proposal.

To exemplify this concept, Figure 4.1 illustrates the uplifting procedure with a simple case (a hypergraph with two 2-hyperedges and one 3-hyperedge uplifted to order 3 with an auxiliary node).

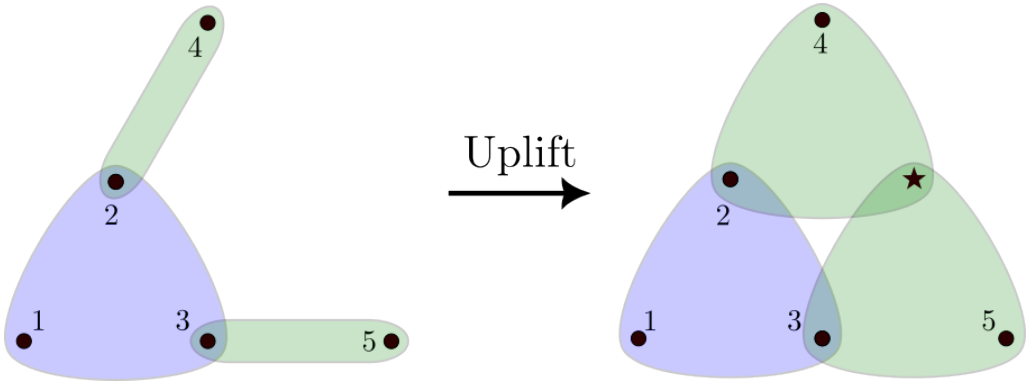


Figure 4.1: Uplift of the hypergraph $H = (\{1, 2, 3, 4, 5\}, \{\{1, 2, 3\}, \{2, 4\}, \{3, 5\}\})$ to $\widetilde{H}$.

The next step is constructing its associated tensor $\widetilde{\mathcal{T}}$, in particular its components $\widetilde{T}_{i_1 i_2 \ldots i_m}$, which will be used in the centrality calculation. In order to do so, one can, naïvely, identify $\star$ as the node $N + 1$ and start filling in the entries of the tensor. There's a caveat, though: as we are considering undirected hypergraphs, the tensors at each order are considered symmetrized. Adding the extra node would provide more permutations to each hyperedge than those originally present. We can avoid this double counting by adding suitable combinatorial factors to the hyperedges which have been uplifted.

Taking this into account, we can define the uplifted tensor as follows,

Definition 4.8 (Uplifted tensor of a hypergraph). *Let $H = (V, E)$ be an unweighted, uniform hypergraph and let $\widetilde{H} = (\widetilde{V}, \widetilde{E})$ be its uplift to order m. The components $\widetilde{T}_{i_1 i_2 \ldots i_m}$ of the uplifted tensor $\widetilde{\mathcal{T}}$ associated to $\widetilde{H}$ are defined as*

$$\widetilde{T}_{i_1 i_2 \ldots augmentation i_m} = \begin{cases} 1 & \text{if} \quad \{i_1, i_2, \ldots, i_m\} \in E, \\ \dfrac{m^\star (m - m^\star)!}{m!} & \text{if} \quad \{i_1, i_2, \ldots, i_m\} \notin E \wedge \{i_1, i_2, \ldots, i_m\} \in \widetilde{E}, \\ 0 & \text{otherwise,} \end{cases} \tag{4.12}$$

where $i_n \in \widetilde{V}$, $n = 1, \ldots, m$ and with $m^\star$ being the number of times the auxiliary node $\star$ was added to the original hyperedge $e = \{j_1, \ldots, j_{m-m^*}\} \subset \{i_1, \ldots, i_m\}$ during the uplifting procedure.

Note that this tensor is weighted (although still non-negative) by construction. An analogous construction could be consider for weighted hypergraph, although we have omitted it for the sake of clarity.

Continuing with the example hypergraph considered in Figure 4.1, we would have

$$T_{\sigma(123)} = 1, T_{\sigma(24)} = 1, T_{\sigma(35)} = 1 \xrightarrow{\text{Uplift}} \widetilde{T}_{\sigma(123)} = 1, \widetilde{T}_{\sigma(24\star)} = 1/3, \widetilde{T}_{\sigma(35\star)} = 1/3, \tag{4.13}$$

where $\sigma(ij) \in \mathfrak{S}_2$, $\sigma(ijk) \in \mathfrak{S}_3$ denote any possible permutation of the indices. This is sensible because, for instance, previously there were two components describing the hyperedge $\{2, 4\}$, and now there are 6 describing $\{2, 4, \star\}$ but with diminished weight.

If the original hypergraph is strongly connected, the addition of a node to some hyperedges will not hinder the connectivity of the resulting hypergraph. Therefore, in that case it is simple to conclude the strong connectedness of the resulting (uplifted) hypergraph.

Lemma 4.9 (Strong connectedness of the uplifted hypergraph). *Let $H = (V, E)$ be a strongly connected hypergraph whose maximum hyperedge size is M and let $m > M$. Then, the uplifted hypergraph $\widetilde{H}$ is strongly connected.*

It is worth noting that, if the original hypergraph is disconnected, the uplift may connect it. This enhanced connectivity is an artifact of the operation, and even though one will get a unique, well-defined centrality for that hypergraph, it might be of interest to carefully check whether that is preferable in the application/system in mind.

An uplifted nuance. Given that the uplift produces uniform hypergraphs, one could now think of apply either HEC or ZEC to the uplifted hypergraph, in order to obtain the centrality score of each node c_i. It turns out that there is a nuance in the uplifting procedure which impedes its usage in the ZEC case, which is not an issue in the HEC case, as we will now see.

Take, for example, a hypergraph H with only size 2 (pairwise) and 3 (triple) interactions. Suppose we uplift it adding an auxiliary node $\star$ once inside each of the pairwise edges. The aim of this procedure boils down to enabling the following rewriting of the "tensor apply" (present in both HEC and ZEC, see Subsection 4.1.2) operation, grouping all the interactions together

$$\sum_{j=1}^{N} a^{(2)}_{ij} c_j + \sum_{j,k=1}^{N} T^{(3)}_{ijk} c_j\, c_k \to \sum_{j,k=1}^{N} \widetilde{T}_{ijk} c_j\, c_k + \sum_{k=1}^{N} \widetilde{T}_{i\star k} c_\star\, c_k + \sum_{j=1}^{N} \widetilde{T}_{ij\star} c_j\, c_\star = \widetilde{T}\mathbf{c}\,\mathbf{c}. \tag{4.14}$$

However, this involves a sum (the pairwise one) where the centrality of $c_\star$ would be involved. It would thus seem that there is a flaw in using the uplift to obtain centralities of the original hypergraph: they would depend on the centrality of the spurious node $\star$. Luckily, in the HEC centrality this is not the case, as we will show that this is an artifact of the procedure which we can get rid of.

To see this, consider applying HEC to an uplifted hypergraph $\widetilde{H}$, obtaining a centrality vector $\mathbf{c} = (c_1, ..., c_N, c_\star)^T$. As discussed before, if $\widetilde{H}$ is strongly connected, $\mathbf{c}$ is positive and unique up to scaling [96]. Therefore, one can always rescale it such

that $c_\star = 1$. This choice solves the previously mentioned apparent contradiction, as the sums before and after the uplift would now coincide.

Moreover, given that the centrality scores we care about are just those of the "real" nodes, one would then just keep the centrality components associated to them, which can then be rescaled again at will (for instance, in order to normalize them). Hence, the initial scaling to achieve $c_\star$ was just a formal consistency check, but it can conveniently be ignored once we have computed the HEC solution.

The ZEC problem. Notice that the reason why we could ignore the aforementioned issue in the HEC case is the fact that if $\mathbf{c}$ is an $\mathcal{H}$-eigenvector, then $\mathbf{c}' \propto \mathbf{c}$ is still a $\mathcal{H}$-eigenvector. This is not the case for $\mathcal{Z}$-eigenvectors: they can't be rescaled and still solve the $\mathcal{Z}$-eigenproblem defined in Subsection 4.1.2 (recall that they are subject to a normalization constraint [97]).

It would seem that there is no use for the combination of uplift and $\mathcal{Z}$-eigenvectors. That turns out not to be the case, if we uplift an already 2-uniform hypergraph to a $(2+m)$-uniform hypergraph, as we will see. From the point of view of computing importance scores this is unnecessary (the ZEC could already be computed in the original hypergraph), but we will see that it plays an important role in the characterization of Perron-like $\mathcal{Z}$-eigenvectors for certain types of hypergraphs. This result will, in turn, feed back into the ZEC centrality quite naturally.

Therefore, from now on we will separate the discussion in two parts: on the one hand, if one starts with a non-uniform hypergraph, its uplift can be used to compute HEC-like centralities. This is the content of the next Subsection. On the other hand, if one starts with a uniform hypergraph, its uplift can shed light on properties of certain $\mathcal{Z}$-eigenproblems. This is an interesting topic although tangential to the topic of centrality measures, which is why it is left aside in Appendix B.

4.2.2 Uplift in $\mathcal{H}$-eigenvectors: spectral centrality in non-uniform hypergraphs

We will now particularize what we have been discussing to the case of $\mathcal{H}$-eigenvectors,. As we mentioned, the main interest of this uplift is the extension of the HEC centrality measure to the case of non-uniform hypergraphs. Given what we know so far, we can already do so.

Definition 4.10 (m-UHEC). *Let $H = (V, E)$ be a strongly connected hypergraph whose maximum hyperedge size is M and let $m \geq M$. The m-Uplifted $\mathcal{H}$-Eigenvector Centrality (m-UHEC) of the hypergraph H consists of the $n = |V|$ components associated to nodes in V of the HEC of the uplift of H to order m.*

For the sake of conciseness and to avoid cluttering the notation, we will from now on refer to the m-UHEC as just the UHEC, with the order being clear by the context, or specified otherwise.

Note that if H is already M-uniform and $m = M$, then the UHEC and standard

HEC vectors coincide. It is straightforward to see that this measure is well-defined in the sense that the UHEC vector is positive and unique (up to scaling), as was the HEC measure.

Theorem 4.11 (Existence and uniqueness of the UHEC). *Under the assumptions of Definition 4.10, the UHEC vector exists and it is unique.*

Proof. Lemma 4.9 guarantees the strong connectedness of the uplifted hypergraph, and the Perron-Frobenius theorem for strongly connected hypergraphs [96] guarantees the existence and uniqueness of its HEC. □

4.2.2.1 The pairwise case

The uplift procedure is not restricted to higher-order networks: one can also apply it to pairwise interaction networks. While in real applications there is no obvious reason why one would prefer it to other, well-established, spectral centrality measures, for us it will be interesting to discuss it as a means of comparison with them: if the centrality outcome after the uplift into a hypergraph was considerably different from the centrality outcome of the pairwise eigenvector centrality, that would have signaled a flaw in our approach.

Consider a pairwise interaction graph $G = (V, E)$ with adjacency matrix $A = (a_{ij})$. Uplift it to $\widetilde{H}^3 = (\widetilde{N}, \widetilde{E})$. Its 3-uniform tensor can be decomposed as

$$\widetilde{T}_{i_1 i_2 i_3} = \begin{cases} a_{i_1 i_2}/3 & \text{if } i_3 = \star \\ a_{i_2 i_3}/3 & \text{if } i_1 = \star \\ a_{i_1 i_3}/3 & \text{if } i_2 = \star \\ 0 & \text{otherwise.} \end{cases} \tag{4.15}$$

With this decomposition, one can rewrite the HEC equation (4.6) as

$$\lambda c_i^2 = \cancel{\sum_{j,k=1}^{N} T_{ijk} c_j\, c_k} + \sum_{j=1}^{N} T_{ij\star} c_j\, c_\star + \cancel{T_{i\star\star} c_\star\, c_\star} = \frac{1}{3} \sum_{j=1}^{N} a_{ij} c_j\, c_\star, \tag{4.16}$$

$$\lambda c_\star^2 = \sum_{j,k=1}^{N} T_{\star jk} c_j\, c_k + \cancel{\sum_{j=1}^{N} T_{\star j\star} c_j\, c_\star} + \cancel{T_{\star\star\star} c_\star\, c_\star} = \frac{1}{3} \sum_{j,k=1}^{N} a_{jk} c_j\, c_k. \tag{4.17}$$

The second of these equations is not actually relevant: it just ties the value of the auxiliary node based on the scores of the rest. Requiring now the centrality of the auxiliary node to be $c_\star = 1$ we end up with

$$\lambda c_i^2 = \frac{1}{3} \sum_{j=1}^{N} a_{ij} c_j, \tag{4.18}$$

which resembles, up to the c_i^2, a weighted, undirected version of the eigenvector centrality equation (2.21). It is therefore natural to compare the centralities ob-

tained through this uplifted measure to those of the standard (pairwise) eigenvector centrality.

We expect to obtain a similar ranking (in the sense of ordering of nodes by importance), although with a lower spread in the actual centrality scores. This is because, loosely speaking, the uplift compresses the centrality scores: the auxiliary node ties every node together, homogenizing the centrality. The most notable thing, is however, the fact that this homogenization can change the actual ranking between the nodes, as can be seen in Figure 4.2.

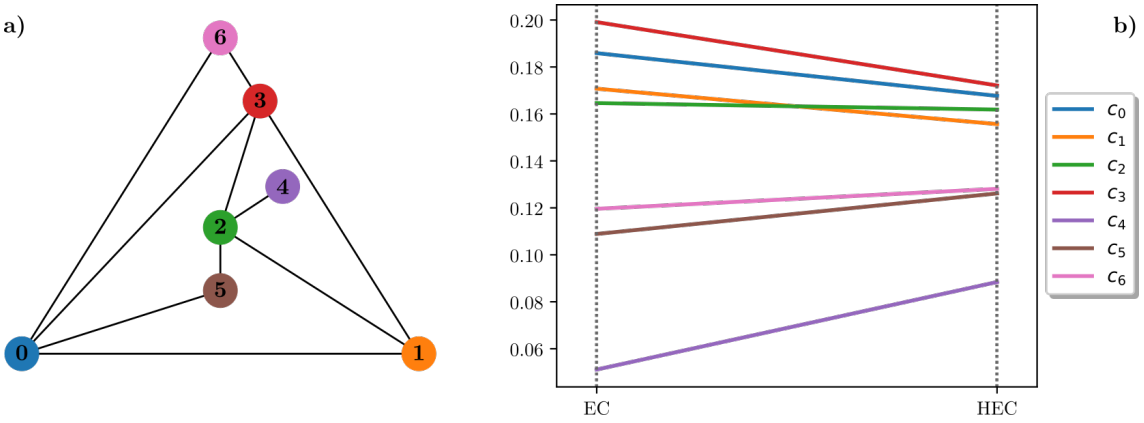


Figure 4.2: Panel **a)** shows the pairwise graph $G = (V, E)$ with 7 nodes whose Eigenvector Centrality (EC) and the HEC of its uplift to order 3 (HEC) are calculated. Panel **b)** shows the centrality values associated to the ℓ_1-normalized centralities in a parallel coordinate plot. The effect of the homogenization can be clearly seen, as well as the crossing in ranking between nodes 1 and 2.

4.2.2.2 Uplifting and projecting

So far we have discussed the way to deal with the hyperedges of size lower than the desired one, by means of uplifting those below it. But in order to truly embrace non-uniform hypergraphs we should also consider an operation bringing hyperedges of higher orders down to the desired one.

The key to this has been hinted at when discussing the Clique motif Eigenvector Centrality, back in Subsection 4.1.2. There, an order k hyperedge is split into all possible pairwise relations, $C(k, 2) = \binom{k}{2}$ of them, between its constituents. In other words, size k edges are projected into sets of size 2 edges. We can think of an analogous process but turning size k edges into sets of size $p < k$ edges.

Definition 4.12 (Projected hypergraph). *Let $H = (V, E)$ be a hypergraph whose maximum hyperedge size is M and let $2 \leq p < M$. Denote the set of hyperedges of size greater than p as E' and denote the set of all p-subsets of every element of E' as S. We define the projected hypergraph hypergraph at order p as*

$$\widehat{H} = (V, \widehat{E}), \quad \widehat{E} = (E \backslash E') \cup S. \tag{4.19}$$

In other words, we can break apart each hyperedge of dimension k into $C(k,p) = \binom{k}{p}$ distinct hyperedges of dimension p.

Notice that, unlike what we did in the uplift case, we can't as of yet define an associated adjacency tensor, as $\widehat{H}$ will generally still be non-uniform. However, given that this operation entails, essentially, a substitution of each higher size edge by a collection of smaller ones, we need to discuss how to assign weights to the smaller ones generated from the projection.

If we follow a similar reasoning to the combinatorial one used in the uplift case (see Definition 4.8), one ends up with nonsensical weight assignments, particularly it can be calculated to be

$$w = \frac{k!}{p!\, C(k,p)} = (k-p)!. \tag{4.20}$$

For instance, an order 4 hyperedge projected would be projected into order 2 hyperedges with weight $w = 2$, hence the resulting hyperedges would have a higher participation than those already at the chosen order.

Instead, we can go back to Benson's work [26], and in particular the CEC calculation, which achieves a sensible projection assigning weights which are the result of counting how many times a pair participates in higher size edges. Our projection aims to generalize this concept, thus the weights come from a similar counting argument (a p-subset's weight will be the number of higher-than-p order edges where the subset participates).

Joining everything together: as we mentioned, the resulting projected hypergraph $\widehat{H}$ might not be uniform, moreover, if we discard orders lower than p we are losing information, as if we uplift hyperedges discarding even higher interactions. For that reason, the key idea in terms of computing centralities is combining both projection of orders higher than a chosen order p and uplifting the lower ones.

Definition 4.13 (p-UPHEC). *Let $H = (V, E)$ be a hypergraph whose maximum hyperedge size is M and let $2 \leq p \leq M$. The p-Uplifted-Projected $\mathcal{H}$-eigenvector centrality (p-UPHEC) is the only positive $\mathcal{H}$-eigenvector of the uniform, weighted hypergraph resulting from:*

1. *Adding an auxiliary node (or more than on as long as they are indistinguishable) to each hyperedge of size $k < p$ and weighting them with their corresponding combinatorial factor.*

2. *Projecting down each hyperedge of size $k > p$ into a set of size p hyperedges, with their corresponding combinatorial factors.*

As in the UHEC case, for the sake of conciseness we will from now on refer to the p-UPHEC as just the UPHEC, where again the order will be clear by the context, or specified otherwise.

It is straightforward to check is the fact that the connectivity of the resulting hypergraph is unchanged.

Lemma 4.14 (Strong connectedness of the projected hypergraph). *Let $H = (V, E)$ be a strongly connected hypergraph and $2 \leq p \leq M$. The hypergraph resulting from uplifting and projecting as in Definition 4.13 is strongly connected.*

And once again, we can easily show the consistency of this measure, as was the case with the UHEC.

Theorem 4.15 (Existence and uniqueness of the UHEC). *Under the assumptions of Definition 4.10, the UHEC vector exists and it is unique.*

Proof. Lemmas 4.9 and 4.14 guarantee the strong connectedness of the hypergraph resulting from projecting and/or uplifting, and the Perron-Frobenius theorem for strongly connected hypergraphs [96] guarantees the existence and uniqueness of its HEC. □

Note that there may be different UPHEC solutions associated to different values of the parameter p. To see this, consider the following example.

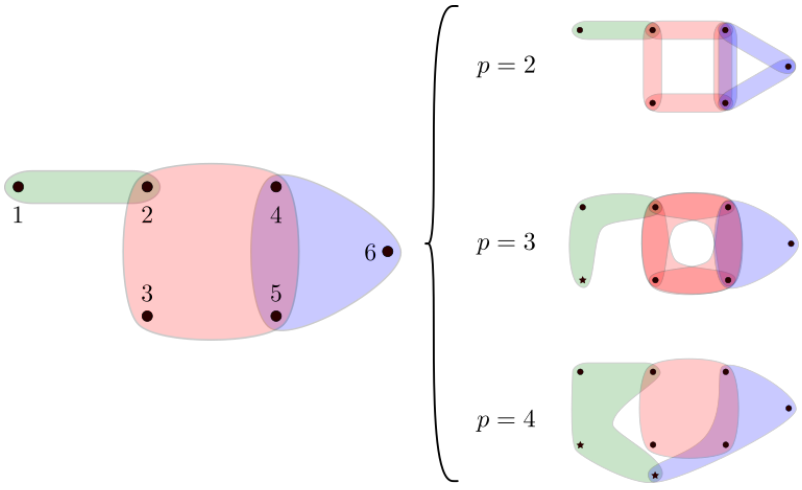


Figure 4.3: Example hypergraph and its three possible uniformizations at $p = 2, 3, 4$.

Example 4.16. *Let $H = (V, E)$ with $E = \{\{1, 2\}, \{2, 3, 4, 5\}, \{4, 5, 6\}\}$, hence $M = 4$. There are three possible UPHEC vectors one can obtain, one for each $p = 2, 3, 4$.*

- *Case $p = 2$: This is equivalent to only considering the projection to order 2.*

$$H' = (V, E'), \quad \textit{with}$$
$$\widetilde{E} = \{\{1, 2\}, \{2, 3\}, \{2, 4\}, \{2, 5\}, \{3, 4\}, \{3, 5\}, \{4, 5\}, \{4, 6\}, \{5, 6\}\}. \quad (4.21)$$

- *Case $p = 3$: In this case we mix the projection of the second hyperedge and the uplift of the first one, therefore computing the HEC of*

$$H' = (V', E'), \quad \text{with}$$
$$E' = \{\{1, 2, \star\}, \{2, 3, 4\}, \{2, 3, 5\}, \{3, 4, 5\}, \{4, 5, 6\}\}. \tag{4.22}$$

- *Case $p = 4$: This is equivalent to only considering the uplift to order 4, i.e. computing the 4-UHEC of*

$$\widetilde{H} = (\widetilde{V}, \widetilde{E}), \quad \text{with} \quad \widetilde{E} = \{\{1, 2, \star, \star\}, \{2, 3, 4, 5\}, \{4, 5, 6, \star\}\}. \tag{4.23}$$

Constructing the respective adjacency tensors and computing their Perron-like $\mathcal{H}$-eigenvector, we get the normalized centrality scores of Table 4.1.

Case	c_1	c_2	c_3	c_4	c_5	c_6
$p = 2$	**0.0929**	0.1802	0.1690	**0.2084**	**0.2084**	0.1412
$p = 3$	**0.0623**	0.1949	0.1943	**0.2060**	**0.2060**	0.1364
$p = 4$	**0.0853**	0.1959	0.1953	**0.1993**	**0.1993**	0.1250

Table 4.1: Centrality scores for each UPHEC case. Cells highlighted in green show the most central nodes in each case, while cells highlighted in red show the least central nodes. A good consistency check is the fact that the centralities of nodes 4 and 5 are identical in either case, as they are indistinguishable in the original hypergraph H.

We can compare this method with the vector centrality one [100], which also distinguishes centralities order by order. The resulting centralities can be found in Table 4.2.

Order	c_1	c_2	c_3	c_4	c_5	c_6
2	0.5	0.5	0.0	0.0	0.0	0.0
3	0.0	0.0	0.0	0.3333	0.3333	0.3333
4	0.0	0.25	0.25	0.25	0.25	0.0

Table 4.2: Centrality scores yielded by the vector centrality measure at each order.

Here we see an example where the new measure improves upon existing ones, as it performs a similar task but it is capable of aggregating information of the whole hypergraph structure into each of the evaluated orders, rather than dismissing those which the nodes do not belong to.

In fact, one can see that one the whole structure is taken into account, in this example there would be no doubt about which is the least important node in the whole network and which are two most important ones. If one were to trust the vector centrality at second order, for instance, one could have been deceived into thinking that the first node is of rather remarkable importance. Moreover, the naïve way to combine these orders (summing the scores of each node) would also lead us to think that node 1 is more important than node 3, for example. It should be clear by now that the non-linear treatment is offering us valuable insights.

Notes on computational complexity: Before moving to real-world applications, we first want to address the computational cost of the algorithms discussed so far.

Firstly, we need to discuss the creation of the tensor, which will have a different complexity depending on whether we are uplifting or projecting a hyperedge. In the case of the uplift, for every hyperedge that has to be uplifted we add the phantom node the necessary times (linear operation). It gets more complicated in the case of projecting, where we needed to compute all the possible combinations of a hyperedge (factorial operation). Let $H = (V, E)$ by a hypergraph, $m \in \mathbb{N}$ the order we want to transform it to. Let $|E| = |E_u| + |E_m| + |E_p|$, where $|E_u|$ is the number of hyperedges that have to be uplifted, $|E_m|$ is the number of hyperedges already at the desired order and $|E_p|$ is the number of hyperedges that have to be projected. Then, the overall number of operations that have to be done to create this weighted tensor is

$$\sum_{e_u \in E_u} (m - |e_u|) + \sum_{e_p \in E_p} m \binom{|e_p|}{m} = |E_u|(m - \bar{e}_u) + \sum_{e_p \in E_p} m \binom{|e_p|}{m}, \tag{4.24}$$

with $\bar{e}_u$ being the average size of hyperedges that have to be uplifted. To compute the Big-O notation we have to choose the worst case scenario, the highest order term. In this case, it will be that associated to the projected edges

$$\mathcal{O}\left(|E| \cdot m \cdot \binom{|e|}{m}\right). \tag{4.25}$$

Once we have created the tensor, we now need to compute the eigenvector corresponding to the largest $\mathcal{H}$-eigenvalue. In order to compute UHEC and UPHEC centralities, instead of creating a new algorithm, we have used a variant of the power method with a weighted tensor (see [105]).

4.2.3 Numerical comparisons

A first attempt to generalize adjacency tensors in a non-uniform context was provided by [106] (and later glossed over by Benson in [26]), which goes by the name hyperedge "blowups" [95]. This method relies on suitably duplicating indices in the adjacency tensor to accommodate to higher order hyperedges, and it has recently been computationally improved so as to avoid its high computational cost when it

comes to the tensor apply operation [95]. However, and as [97] already points out, there is some indeterminacy in this approach.

We will nevertheless consider the original (and only) proposal [106] which, given a hyperedge $e = \{v_1 \dots v_s\}$ with $2 \leq s \leq m$ nodes (where m is the maximum cardinality of the hyperedges), assigns it the m-uniform adjacency tensor components

$$a_{i_1 \dots i_m} = \frac{s}{\alpha} \quad \text{where} \quad \alpha = \sum_{p_1,\dots,p_r=1}^{N} \frac{m!}{p_1! p_2! \dots p_s!}, \tag{4.26}$$

and $i_1, \dots, i_m$ are chosen in all possible ways from $\{v_1, \dots, v_r\}$. The construction of this tensor is already disadvantageous. Time complexity of this uplifting method can be directly found by the intuitive idea behind it. Let's say we want to uplift the hyperedge e to order m. To do so, we will need all the possible combinations of adding each node to it, until we reach the desired order. Increasing the order by 1 would take $|e|$ operations (add each node to the hyperedge once). Increasing the order by 2, we would need to do $|e^2|$ operations (the mentioned before, and for each new hyperedge constructed, add each of the original nodes). It's straightforward that the time complexity we are talking about is $\mathcal{O}\left(|e|^{m-|e|}\right)$ for each hyperedge to be uplifted. Nevertheless, this time complexity can be reduced through dynamical optimization to $\mathcal{O}\left(|e|(m-|e|)\log m\right)$. The method proposed in this paper to uplift a hyperedge involves far fewer operations, having $\mathcal{O}\left(m-|e|\right)$ for each hyperedge, as the only thing it is being done is adding a new node the necessary times. Moreover, this alternative uniformization does not include a notion of projection, which is why we have to supplement it with ours if one is interested in checking intermediate orders.

We now want to give a flavour of the difference between the different tensorial methods discussed throughout this manuscript, namely: the standard HEC (equation 4.5), the UPHEC (Definition 4.13) and the alternative uniformization method (equation 4.26), at each of the different orders present in a hypergraph[2].

4.2.3.1 Real-world hypergraph datasets:

As further evidence of the interest and usefulness of our method, we now want to take several real world hypergraph datasets into consideration, analyzing three different points: firstly, how do the two hypergraph uniformizations discussed (our UPHEC and blowups [106, 95]) compare against the original, order-by-order analysis of the hypergraph put forward by Benson [26]. Secondly, what is the difference between both uniformizations in these real cases. And thirdly, even within either of these uniformizations, one needs to choose at which order to perform the analysis (uplifting lower orders and projecting higher ones in our method, "blowing up" lower orders and also projecting higher ones in the blowup one.

It is important to note that the besides the figures shown in the present manuscript, which have been picked due to their clarity and aid in the exposition, we have perform

[2]We will not be discussing the ZEC here, as it was already done in [26] and it does not have an UPHEC analogue, as discussed throughout the text.

a wider analysis of more hypergraphs (all of them freely available within the XGI library [107]).

To start things off, let us consider two hypergraphs: a quintessential one, the `tags_ask_ubuntu` dataset [108], also used in [26] to showcase the CEC, ZEC, and HEC proposals, and the `hypertext-conference` one [109]. The former contains information about interactions within the *Ask Ubuntu StackOverflow* forum. Specifically, it can be seen as a hypergraph where nodes represent tags and hyperedges between tags represent questions asked marked with those tags. The latter contains data gathered during the *ACM Hypertext 2009* conference, pertaining the interactions between its participants.

Some basic statistics of these hypergraphs (after pre-processing them with the XGI library [107] in order to remove isolated nodes, singleton edges, etc) can be observed in Table 4.3. Note that when studying each uniform order as isolated some nodes will become disconnected if they have no such interactions.

	`tags_ask_ubuntu` [108]			`hypertext-conference` [109]		
Order	Nodes	Hyperedges	Size of LCC	Nodes	Hyperedges	Size of LCC
2	2714	28134	89.84%	113	2103	100%
3	2821	52282	93.38%	105	302	92.92%
4	2722	39158	90.10%	11	12	9.73%
5	2564	25475	84.87%	8	7	7.08%
6	-	-	-	8	4	7.08%
Complete	3021	145053	100%	113	2434	100%

Table 4.3: Number of nodes and hyperedges at each order of the hypergraphs, as well as the size of the Largest Connected Component (LCC) containing them compared to the total. There is a small discrepancy between the sum of hyperedges at each order and the total number, due to the fact that some hyperedges become disconnected if we remove other orders and therefore they won't appear in the LCC

The natural way to compare rankings is by means of some correlation measure which only takes into account the ordinal correlation between the entries (i.e. their position within the ranking) rather than their actual magnitudes. One of the best known examples of this measure is Kendall's tau correlation coefficient ($\tau \in [0, 1]$, where the closer to 1 the more correlated), which we will compute between every pair of rankings.

Before showing the actual results, we should mention that in order to compare two rankings, they must contain the same number of elements. However in the uniformized vs non-uniformized cases this is not the case (the non-uniformized, i.e. standard HEC versions only keep the LCC with those interactions). For that reason

we have chosen to fill the empty entries with a zero value, as they do not participate in such order. It is here that we can already glimpse at the issue with the standard, non-uniformized HEC: if we look at Table 4.3 we can see that, while this may be sensible in the `tags_ask_ubuntu` dataset, in the `hypertext-conference` the LCC of order beyond 3 is a minuscule part of the entire hypergraph. For this reason, the ranking will be localized around those nodes, yielding $\tau \approx 0$.

The results of the comparison between each of the rankings are shown in Figure 4.4. At this stage, we will focus on the first of the questions described before: the comparison of either uniformization with the non-uniform approach per order.

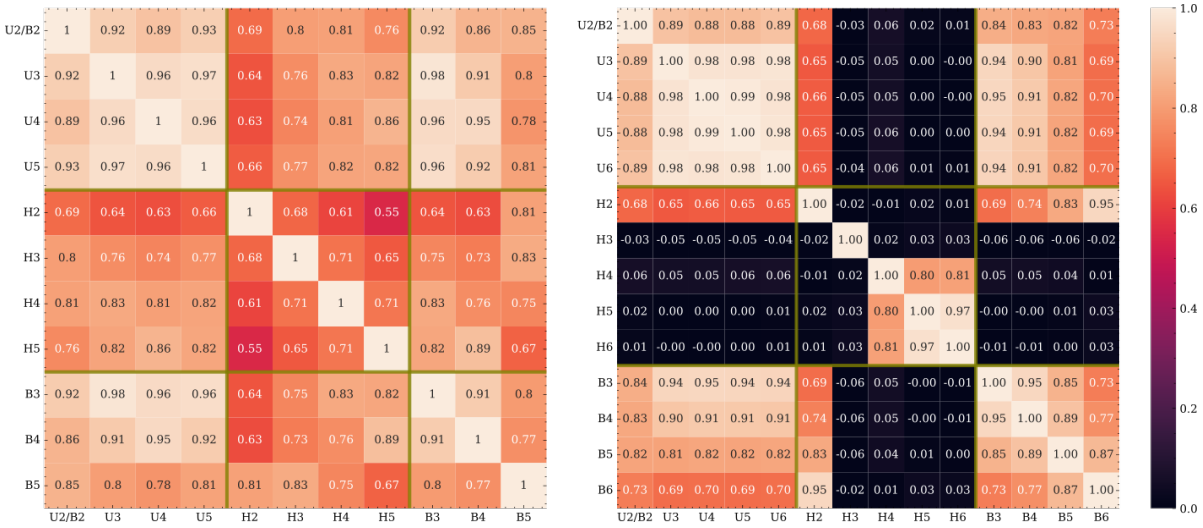


Figure 4.4: Kendall's tau correlation coefficient between the whole rankings obtained in each of the methods, for the `tags_ask_ubuntu` (left) and `hypertext-conference` (right) datasets. Methods are labelled as U2, U3, U4, U5 for the UPHEC case; H2, H3, H4, H5 for the HEC at each order, and B3, B4, B5 for the blowup uniformization discussed above. U2 and B2 are equal as they only include the projection, and are therefore shown together.

In the `tags_ask_ubuntu` dataset, the four standard HEC measures among themselves have the lowest correlations. The lowest correlation of the whole Figure is actually that between these 2nd and 5th orders. This is product of the fact that the uniform hypergraphs at each order have little to do with each other, they each describe a portion of the whole.

As we advanced before, this is much more drastic in the `hypertext-conference` dataset: there almost every correlation yields a number close to zero, except for the one between orders 5 and 6, as the LCC's of these orders share the same 8 nodes (see Table 4.3).

This analysis makes it clear that there is a need for uniformized versions of the HEC centrality, as the order-by-order study of hypergraphs clearly lacks a cohesive description of the whole[3].

[3]For a visual analogy, check the cover of the first edition of "Gödel, Escher, Bach: an Eternal

Having dealt with the question of uniform measures versus non-uniform ones, we now shift our focus to the other two problems: the comparison between uplift and blowups, and the order to inspect. In order to get a better understanding, we supplement the previous examples with other four real hypergraph datasets, also available in XGI, whose most basic statistics (this time without an order-by-order overview) after preprocessing are summarized in Table 4.4.

Dataset	Nodes	Hyperedges	Max. order	$\langle\tau_{UU}\rangle$	$\langle\tau_{BB}\rangle$
`tags_ask_ubuntu` [108]	3021	145053	5	0.960	0.825
`hypertext-conference` [109]	113	2434	6	0.982	0.844
`contact-primary-school` [111]	242	12704	5	0.962	0.905
`contact-high-school` [112]	327	7818	5	0.946	0.863
`sfhh-conference` [113]	403	10541	9	0.918	0.748
`diseasome` [114]	516	314	11	0.724	0.590

Table 4.4: Number of nodes, hyperedges and maximum order of each hypergraph (after removing isolated nodes and duplicated edges and keeping the LCC), as well as average Kendall-tau coefficient between the UPHECs $\langle\tau_{UU}\rangle$ and the blowups $\langle\tau_{BB}\rangle$ at each order between 3 and the respective maximum order.

For each of these hypergraphs, we have computed the correlation between the UPHEC and blowup+projection measures in Figure 4.5, now ignoring the non-uniform measures for ease of visualization[4], as well as the the top-most column and left-most row (the "U2/B2" ones in Figure 4.4, for they correspond to only projecting any higher-order interaction to pairwise ones, and computing the HEC of the corresponding, uniform hypergraph. As there are no uplifts nor blowups, there is no distinction on the uniformization used, which is why we choose to ignore them at this point.

We can draw the following conclusions from Figures 4.4 and 4.5, with respect to both uniformization procedures.

Firstly, the average correlation between the same type of uniformizations at different orders (i.e. the U-U, B-B quadrants) is always higher in the uplift than in the blowup cases. For reference, the average value of off-diagonal correlations in those two quadrants, for each hypergraph, are shown in Table 4.4. In that regard, it is interesting to look at in the `sfhh-conference` example: the lowest correlation between any two UPHEC measures is found to be around 0.88, while the lowest in the blowup uniformization is around 0.55.

Golden Braid", by Douglas R. Hofstadter [110].

[4]In the `diseasome` case we have also restricted the computation to orders up to 9 for better visualization.

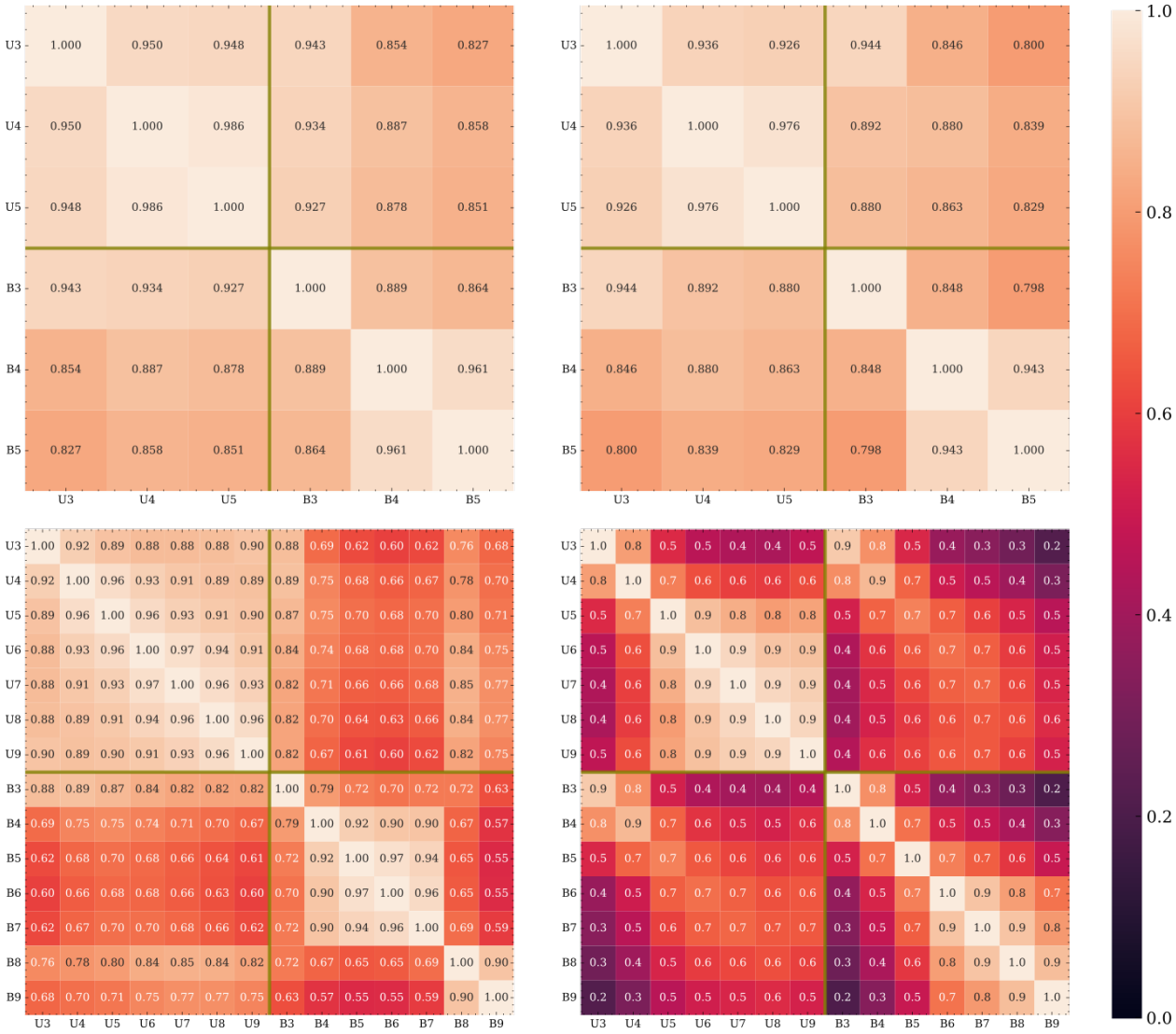


Figure 4.5: Kendall's tau correlation coefficient between the whole rankings obtained by the UPHEC and blowup uniformizations (with the same labelling conventions as in Figure 4.4) for the `contact-primary-school` (top-left), `contact-high-school` (top-right), `sfhh-conference` (bottom-left) and `diseasome` (bottom-right) datasets.

Moreover, one can clearly see that the higher the order inspected, the more disparity. This is pointing out to the fact that, while in lower orders the projection part of the measure (which is the same in both the UPHEC and in the blowup uniformization) is evening the rankings, when we focus on the highest order (thus only having vanilla uplift and blowup, no projection) the blowup is computing something slightly different. In that sense, this seems to confirm the claim in [97] about the blowup uniformization and the fact that it contains a degree of arbitrariness in the augmentation, something which indeed observe.

Apart from the choice of uniformization procedure, we wanted to understand the

implications of the choice of order at which to inspect the hypergraph. Focusing on the UPHEC method, what we can see is that in most examens the choice is basically irrelevant: once we take into account every level of interaction (either through projection or uplift), a centrality unison emerges, something we can clearly see from $\langle \tau_{UU} \rangle \approx 1$. Nevertheless, the correlation is better at higher orders, meaning that the more uplift and less projection, the more agreement in the description of the overall hypergraph.

At this point it is important to also consider the computational cost of each of these methods. As we have discussed, ideally one would want to compute the centrality with UPHEC at the highest order available. However, doing so is rather costly, hence it might be preferable to have a balance between uplift and projection, staying therefore at intermediate orders. Alternatively, and if computational efficiency is a necessity, one could use the method proposed in [95], which achieves a remarkable speed increase in the computation of the blowup, turning an $\mathcal{O}(N^M)$ problem into one being polynomial in M, the maximum order.

Apart from the full ranking comparison, it is often interesting to understand how does the correlation change when we contrast the top K nodes obtained with a method with their corresponding ranking according to another method, as we increase the amount K of nodes sampled.

Given the amount of possible comparisons (12 in the case of UPHEC-UPHEC, 16 in the cases of UPHEC-HEC, etc), we decided to filter out most of them in order to present a meaningful figure. In particular, for each measure comparison we have chosen to keep at most correlations: the correlation reaching the highest maximum, the correlation reaching the lowest minimum, and the two correlations whose average is minimum and maximum. We feel that these conditions will provide us with a set of correlations which can convey more information (in the sense of most similar and dissimilar rankings). The resulting plot is displayed in Figure 4.6, and the unfiltered one can be found in the open repository available at https://github.com/LaComarca-Lab/non-uniform-hypergraphs.

We see that despite some initial fluctuations around $K = 100$, most correlations tend to increase or stabilize, converging to their respective values shown in Figure 4.4. We also notice that in most cases the minimums are reached in pairs, e.g. U2 and U4 are not very correlated with each other in Subfigure 1**a** in either direction, which is rather sensible.

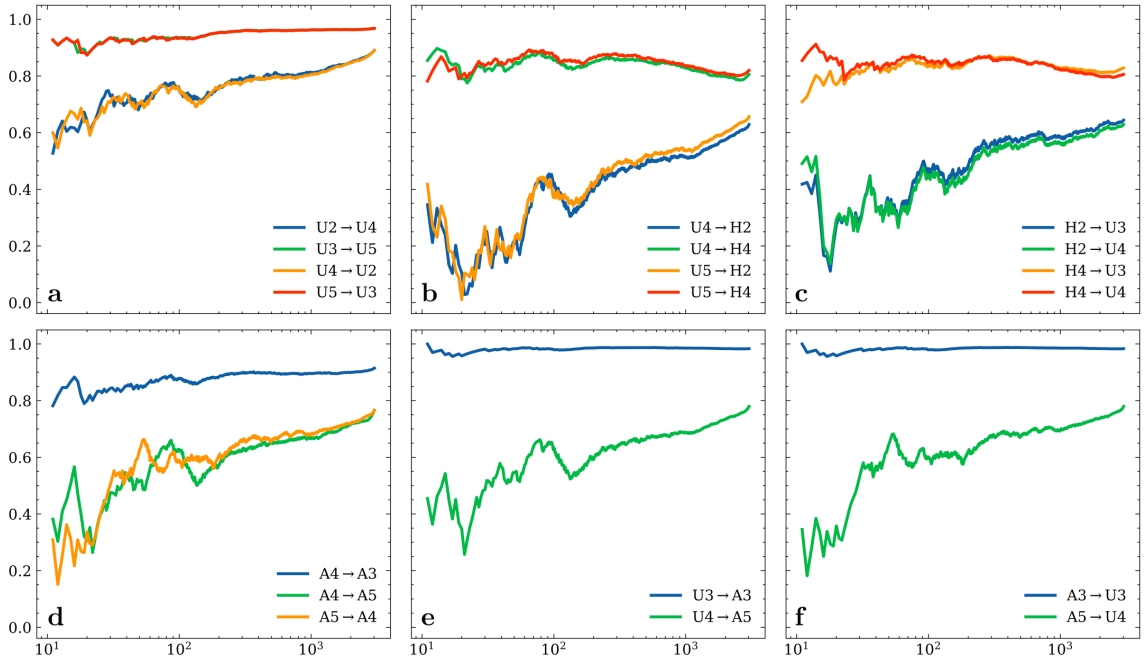


Figure 4.6: Kendall's tau correlation coefficient (log scale) between the top K ranked nodes from one method with the same nodes from another method on the Ask Ubuntu co-tagging dataset. Panels **a**, **b**, **c**, **d**, **e** and **f** represent the UPHEC-UPHEC, UPHEC-HEC, HEC-UPHEC, Alternative-Alternative, UPHEC-Alternative and Alternative-UPHEC, respectively.

4.3 Directed higher-order interactions and beyond

In this Section we want to begin our quest to generalize hypergraphs, to include some notion of directedness. So far in the literature, the notion of directed hypergraphs has been restricted to a special type of higher order interaction, especially amenable to a set-theoretic description [115]. These interactions require a clear division between the nodes in the hyperedge between "tail" (or input) and "head" (or output) nodes. Our first task will be breaking free from this constraint and understanding how we can build directed interactions from other points of view. What's more, even within the set-theoretic framework we will see that it is easy to find new types of structures. The wide variety of hypergraph types requires a new adjective to describe them all, beyond traditionally directed hypergraphs, which is why we chose to call them heterogeneous hypergraphs.

Once this issue has been dealt with, we will give some examples of where this description could be useful, and we will end this Section with a technical interlude regarding an operation which is especially relevant in directed graphs, the transposition of the adjacency matrix, and how to extend it to higher order situations.

4.3.1 Algebra, topology and set theory

Let us start from the very beginning: upon opening any textbook or scientific publication on graph theory, one is most likely going to find the definition of a graph to be $G = (V, E)$, where V is the set of nodes and $E \subseteq V \times V$ is the set of edges [9, 32]. The distinction between directed and and directed edges lies, in this context, on whether the edges $e \in E$ are ordered or unordered pairs of nodes. Notice that this is a set theoretic point of view: a graph can also be understood from a topological point of view (nodes and connections between them) or from an algebraic point of view (e.g. adjacency matrix), see Figure 4.7.

These different approaches are so important that they even have names of their own (topological graph theory, algebraic graph theory), as they make use of different techniques, tools and results. In standard graph theory these three perspectives complement each other, for standard graphs (regardless of directionality) are equivalently described in either.

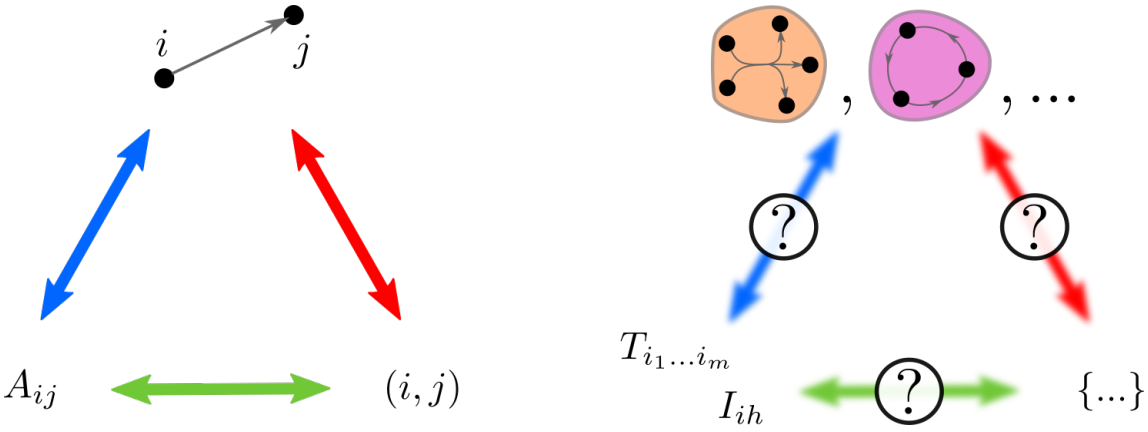


Figure 4.7: Three different points of view: topological (up), algebraic (down-left) and set-theoretic (down-right). In standard directed graphs (left triangle) the translation between the three perspectives is clear. In heterogeneous hypergraphs there is a vast landscape of possibilities, which makes the dictionary between any two perspectives unclear at best, otherwise impossible.

Starting from the set-theoretic viewpoint, the simplest idea is allowing each hyperedge to have multiple "input" nodes, and multiple "output" nodes. In [115] this perspective is fleshed out: a directed hyperedge is a tuple $E_k = (T(E_k), H(E_k))$ where $T(E_k) \subseteq V$ is the "tail" of the hyperedge, while $H(E_k) \subseteq V$ is the "head" of the hyperedge, and such that[5] $H(E_k) \cap T(E_k) = \emptyset$. We will elaborate on this type of hypergraph structures on Subsection 4.4.2. See Figure 4.8 for an example.

This point of view is advantageous if the application in mind can leverage the set-theoretic nature of the hypergraph, or if the directed incidence matrix I_{ij} of the

[5]According to [115], either the head or the tail set could be empty although we will assume from here on that they aren't.

hypergraph can be involved. However, if that is not the case, then this description of hypergraphs might not be suitable for our problem.

Furthermore, even when within set-theoretic descriptions of higher order interactions, there are plenty of situations that are left out by the input-output paradigm: one could think of a higher order interaction where there are some input nodes, some intermediate nodes, and some output notes (see edge E_4 in Figure 4.8 for an example). In that sense we not only generalize the amount of nodes in the end points of an edge, but we also generalize the amount of stages within a single interaction. This may be relevant, for instance, to represent complicated processes such as computer programs with different functions and modules with intermediate stages.

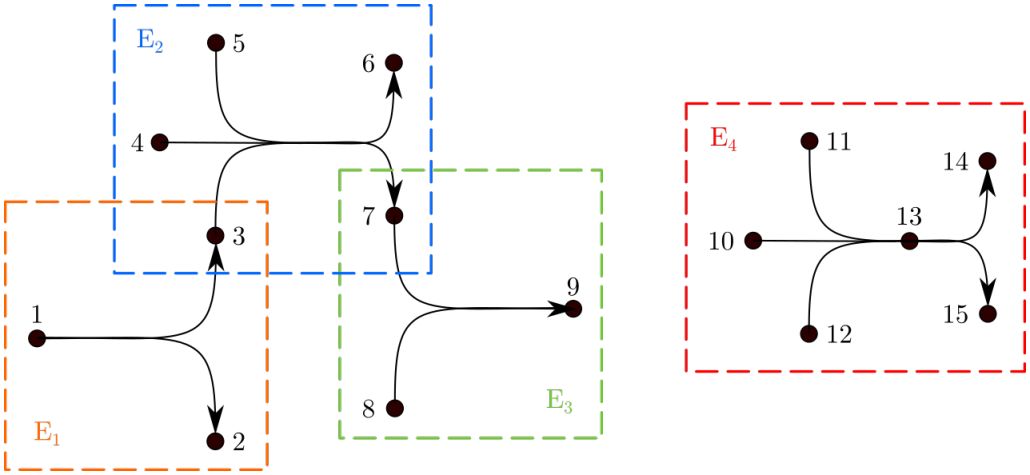


Figure 4.8: Example of a set-theoretic, directed hypergraph $H = (V, \{E_1, E_2, E_3, E_4\})$. The first three edges $E_1 = \{\{1\}, \{2, 3\}\}$, $E_2 = \{\{3, 4, 5\}, \{6, 7\}\}$, $E_3 = \{\{7, 8\}, \{9\}\}$ are standard in directed hypergraphs. The last edge $E_4 = \{\{10, 11, 12\}, \{13\}, \{14, 15\}\}$ is also sensible from a heuristic and set-theoretic point of view, but it does not fit in the input-output scheme. And even though we could reduce it to two edges (one with 13 on its head, one with it on its tail), that defeats the purpose of higher order interactions: the same could be said about any hyperedge, it can be reduced to its pairwise constituents.

This situation is reminiscent of the division between simplicial complexes and hypergraphs: the former is a more restrictive type than the latter, however if we are interested in a problem where its topological features and tools (e.g. homology) can be exploited [53], then it is advantageous to discuss it on its own. Here we can see that if the directedness is present in a binary fashion (input-output), these structures are worth considering, but we should keep in mind that there are other possible ways of non-undirectedness in the interactions of a hypergraph.

From a spectral centrality point of view, the directed hypergraph perspective is rather limiting, as the fundamental object in its study is the adjacency tensor and their spectral properties. While we can work out a way to successfully describe the adjacency tensor of these directed hypergraphs (see the next section), we can get more interesting structures if we start from an algebraic point of view.

What we mean by this is, we will consider the adjacency tensor to the the *fundamental* object describing the structure of a directed, uniform hypergraph, and we will from then make sense of the topology of the hypergraph. And, in certain cases, this will also provide a set-theoretic description, but there will be cases where no set-theoretic description is available. Hence, in the case of uniform hypergraphs, the set-theoretic point of view is a subset of a larger class of hypergraphs, which we call "heterogeneous". And even though we arrived at them via spectral centralities, it is worth noting that they are more ubiquitous.

Having said this, the space of possible heterogeneous, m-uniform hypergraphs is too vast (an adjacency matrix representing a directed graph can have N^2 different entries, and analogously an adjacency tensor can have N^m different entries), and its analysis is therefore not possible in all generality.

We will soon discuss some specific types of directedness depending on their tensorial representations. In that sense, and bringing back the terminology introduced in the undirected case, we will no longer have tensor components (i.e. hyperedges) symmetric under all possible permutations $\sigma(i_1 i_2 ... i_m)$. We will then show how these types of directedness and provide natural definitions for their transposition and strong connectedness (if any).

4.3.1.1 Applications and examples of heterogeneous interaction systems

The possibility of studying systems with heterogeneous interactions in the above-mentioned sense seems so far like a mathematical pastime, bearing no connection to any real system, however that is far from reality. In fact, the original paper on directed hypergraphs [115] already describes some use cases, and the need to describe heterogeneous interactions was already recognized more than a decade ago [116].

Here we briefly review some of the applications described in both references and we add some more which we think are of relevance:

- Biochemical reactions. Online, open databases such as [117] or [118] can be mined in order to construct different directed hypergraphs where edges represent chemical reactions or metabolical pathways between metabolites, respectively [116]. We performed this mining task, and the former example is later showcased in Subsection 4.4.2.

- Citation networks. A classic example of networks and hypergraphs is that of authors as nodes, linked together if they are coauthors in a paper. In the pairwise case the coauthorship is established between any two authors if they wrote a paper together, possibly with others. In the undirected hypergraph case the paper itself is a single hyperedge between all coauthors. But this modelling dismisses the order within the list of authors, a key piece of information in many fields. This ordering can be introduced in the heterogeneous hyperedge case, but not in the directed case.

- Urban transit. In [115] it is argued that there are many mobility networks which can be abstracted as directed hypergraphs, where some nodes are stops

within transportation lines (e.g. metro), and thus connected via pairwise edges representing said lines, while other nodes serve as the actual transit stops, which individuals need to traverse by walking to change lines [119]. These are represented as directed hyperedges.

- Routing/delivery transportation. It is relatively common for transportation businesses to have their trucks repeat a predefined path (e.g. to supply the same stores periodically). This has traditionally been modelled by dividing such paths into their pairwise components, however with heterogeneous hypergraphs we could consider the path itself to constitute a heterogeneous hyperedge, and encode it in the adjacency matrix as an asymmetric tensor component.

  A more restrictive, though more tractable version of this is the case where all paths are of the same length k. This is then related to the k-step eigenvector centrality originally put forward in [120] in the $k = 2$, undirected case, which we will extend in Subsection 4.4.3.

- Online social interactions. Hypergraphs can be constructed from online forum data, where nodes represent users and hyperedges represent threads where they participate. Different amounts of participation can be encoded in the adjacency tensors via asymmetric tensor components, leading to a heterogeneous hypergraph.

- Propositional logic and relational databases. In [115] different such hypergraphs are considered, with the most prominent type abstracting relational databases, where nodes represent propositions, with directed hyperedges linking a series of causes to their consequences. Notice that here the projection of the hypergraph to its pairwise constituents is losing key information (one needs all propositions to be fulfilled at once in order to prove their consequence).

It should be noted that we will not attempt to construct all of these hypergraphs, as this manuscript is of a theoretical nature, and we moreover are interested in those where spectral centrality measures can be computed (for instance, in propositional hypergraphs the avoidance of circular arguments renders eigenvector-like centralities impossible due to the lack of strong-connectivity). Nevertheless we will exemplify our methods with some real hypergraphs in the next section.

4.3.2 The transposition operation

Before moving on to discuss spectral centralities in special heterogeneous hypergraphs, there is a technical detail which must be dealt with in all those cases.

Recall the standard eigenvector centrality of a graph, equation (2.21). There, the transposition step is quite natural, either from a topological point of view (inverting the orientation of each edge), from a set-theoretic point of view (directed edges are ordered sets, transposition means inverting such order), or from an algebraic point

of view (the matrix transposition maps components of a matrix into those of the transposed matrix).

For this reason, computing the eigenvector centrality of a graph is equivalent to computing the Perron eigenvector of the "topologically transposed" or "set-theoretical transposed" graph. In some sense, there is, as we mentioned before, a clear connection between topology, set-theory and algebra. In the case at hand, said the connection is "broken", or rather, it is non-unique. Therefore, in order to make sense of the transposition, we need to establish "by hand" a clear connection between them.

Our approach is an algebraic one, something which is quite reasonable taking into account that the $\mathcal{H}$-eigenproblem is algebraic in nature. With that in mind, we can think of the transposition as a map from the space of tensor components to itself

Definition 4.17 (Transposition of an adjacency tensor). *Let $\mathcal{T} \in \mathbb{R}^{[m,N]}$ be an adjacency tensor and σ a permutation of m elements. The transposition of the tensor is defined, component by component, as $(T_{i_1...i_m})^t = T_{\sigma(i_1...i_m)}$, where $\sigma(i_1...i_m)$ is the permutation of the indices $\{i_1, ..., i_m\}$.*

Essentially, it is a choice of permutation of the indices involved, which in the matricial case is unique, i.e. $(A_{ij})^t = A_{ji}$. In the tensorial case this choice is not unique, but we can make it such that it is mathematically sensible for our problem, depending on the type of directedness we are considering.

Knowing what the transposition looks like from an algebraic point of view, we can now connect it to topology/set-theory.

Definition 4.18 (Transposed hypergraph). *The transposed hypergraph H^t associated to the hypergraph H is the hypergraph induced by the transposed adjacency tensor $(\mathcal{T})^t$ of H.*

What the final structure (the transposed hypergraph) will look like can't be generally assessed, instead we will need to consider appropriate transpositions for each directed case, and that will force upon us the topological/set-theoretic nature of the transposed hypergraph.

What does this entail for the already established, undirected case? Our algebraic definition of a transposition in terms of index shifts is in complete agreement with what's been done so far for undirected hypergraphs. There, the transposition step is ignored altogether. And indeed, in an undirected hypergraph we have the exceptional property that $T_{i_1...i_m} = T_{\sigma(i_1...i_m)}\, \forall \sigma$, hence no matter which transposition rule we choose the outcome would be the same.

4.4 Spectral centralities of heterogeneous hypergraphs

The study of spectral centralities in centrality in directed hypergraphs poses, as discussed above, a conceptual problem: what kind of transpositions are more mathematically coherent?

It is not possible to answer it in all generality, for all possible, unconstrained tensors $\mathcal{T}$. For that reason, we will be giving sensible definitions to the transposition of specifically structured tensors. These will represent either newly defined ones (cyclical hypergraphs and k-step hypergraphs) or more well-known ones (directed hypergraphs).

Also, depending on the constraints we place on the available hyperedges we will be able to give more specific details about the connectivity of the heterogeneous hypergraph. In that regard, it is important to keep in mind that the strong connectivity requirement for the existence and uniqueness of HEC is not a requirement on the hypergraph H, but on the transposed hypergraph H^t.

4.4.1 Cyclical hypergraphs

In this particular kind of heterogeneous hypergraph we restrict hyperedges to have a *cyclicity* permutation symmetry, as defined below.

Definition 4.19 (Cyclical hyperedge). *Let σ_{odd} and σ_{even} be the set of all even and odd permutations of m elements, respectively. A cyclical hyperedge between nodes $\{i_1, \dots, i_m\}$ corresponds to*

$$T_{\sigma(i_1,\dots,i_m)} = T_{\sigma'(i_1,\dots,i_m)},\ \forall \sigma, \sigma' \in \sigma_{\text{odd}} \quad \textit{and} \quad T_{\sigma(i_1,\dots,i_m)} = T_{\sigma'(i_1,\dots,i_m)},\ \forall \sigma, \sigma' \in \sigma_{\text{even}} \tag{4.27}$$

For example, $T_{123} = T_{312} = T_{231}$ and $T_{132} = T_{213} = T_{321}$ correspond to the two possible cyclical hyperedges between nodes $\{1, 2, 3\}$. Note that this type of heterogeneous hyperedges are already not possible to capture with a set-theoretical description (there is no split heads/tails).

A sensible definition for the transposition of the tensor in this scenario is mapping even permutations to odd permutations and viceversa by totally reversing the order of the indices

$$(T_{i_1\dots i_m})^t = T_{i_m\dots i_1}. \tag{4.28}$$

In this type of hypergraph, the "tensor apply" operation (Definition 4.2) required for centrality calculations reads

$$\left[(\mathcal{T})^t \mathbf{c}^{m-1}\right]_{i_m} = \sum_{i_1,\dots,i_{m-1}=1}^{N} T_{i_m\dots i_1}\, c_{i_1} \dots c_{i_{m-1}}. \tag{4.29}$$

Let us now address strong connectivity in regards to the transposition operation. In this case, a cyclic hyperedge in H translates into a directed cycle C_m in G^M, therefore we do not have to worry about the transposition in order to examine the connectivity, as either orientation of a directed cycle is strongly connected.

In principle one can always study the HEC of a heterogeneous, k-uniform hypergraph, provided it is strongly connected, however this will yield no new results in comparison with its undirected counterpart, where $T_{\sigma(i_1\dots i_m)} = T_{\sigma_{even}(i_1\dots i_m)} + T_{\sigma_{odd}(i_1\dots i_m)}$. The reason for this is the fact that (4.29) does not distinguish between the cycle orientations, hence it yields uninteresting results.

While this particular kind of heterogeneous hypergraph is not interesting to study from a spectral centrality point of view, its existence and possible applications to model systems have yet to be unveiled.

4.4.2 Directed hyperedges

We will now consider a directed hypergraph. This is still too broad to tackle, we need to narrow it down further. But before doing so, let us introduce some definitions from set-theoretic hypergraphs.

Definition 4.20 ($\mathcal{B}$- and $\mathcal{F}$-hyperedges [115]). *A backward hyperedge, or simply $\mathcal{B}$-hyperedge, is a hyperedge $e = (t(e), h(e))$ with $|h(e)| = 1$. A forward hyperedge, or simply $\mathcal{F}$-hyperedge, is a hyperedge $e = (t(e), h(e))$ where $|t(e)| = 1$.*

It is common to refer to $h(e)$ as the "head" of the hyperedge, and to $t(e)$ as its "tail" [115]. Notice that any hyperedge $e = (t(e), h(e))$ not belonging to either of them (i.e. $|t(e)|, |h(e)| \neq 1$) can be converted into a $\mathcal{B}$-hyperedge and a $\mathcal{F}$-hyperedge by splitting it placing a node between them (although from the point of view of centrality this is not an appropriate operation, for it creates new nodes with their own centrality). A hypergraph consisting of just $\mathcal{B}$-hyperedges ($\mathcal{F}$-hyperedges) is called a $\mathcal{B}$-hypergraph ($\mathcal{F}$-hypergraph).

Directed hypergraphs can be encoded as a series of adjacency tensors whose components satisfy certain symmetry constrains (this has already been proposed by [121]). Each directed hyperedge contributes to the corresponding adjacency tensor as follows:

- A backward hyperedge $E = (\{i_1 \dots i_{m-1}\}, \{j\})$ corresponds to tensor components
$$T^{(m)}_{\sigma(i_1\dots i_{m-1})j}, \tag{4.30}$$
for any permutation σ.
- A forward hyperedge $E = (\{i\}, \{j_2 \dots j_m\})$ corresponds to tensor components
$$T^{(m)}_{i\sigma(j_1\dots j_m)}, \tag{4.31}$$
for any permutation σ.
- A general hyperedge $E = (\{i_1 \dots i_{m'-1}\}, \{j_{m'} \dots j_m\})$ corresponds to tensor components
$$T^{(m)}_{\sigma_1(i_1\dots i_{m'-1})\sigma_2(j_{m'}\dots j_m)}, \tag{4.32}$$
for any permutations σ_1, σ_2.

From an algebraic point of view, the transposition of such a tensor corresponds to the index transposition

$$(T^{(m)}_{i_1\dots i_{m'-1}j_{m'}\dots j_m})^t = T^{(m)}_{j_{m'}\dots j_m i_1\dots i_{m'-1}}, \tag{4.33}$$

for each tensor $T^{(2)}, T^{(3)}, ..., T^{(M)}$. The following result is then straightforward.

Proposition 4.21 (Transposed directed hypergraph). *The transposed hypergraph H^t of a directed hypergraph $H = (V, E)$ is the result of interchanging $h(e)$ and $t(e)$ in each hyperedge $e \in E$.*

As we will see, the distinction between the two types of hyperedges will have dramatic consequences in centrality computations.

4.4.2.1 $\mathcal{B}$-hypergraphs

The first case of directed hypergraphs we will consider is that where only $\mathcal{B}$-hyperedges are present, with a fixed number of nodes per hyperedge (i.e. m-uniform). In this case the transposition reduces to

$$(T_{i_1\dots i_{m-1}j})^t = T_{ji_1\dots i_{m-1}}. \tag{4.34}$$

This makes it perfectly suitable for the HEC centrality calculations, as the free index in the sum is the first one, and the sums run equally over the remaining indices. It is important to notice that the $\mathcal{H}$-eigenvector one needs to compute is that corresponding to the transposed hypergraph, which will be the one whose connectivity is called into question.

The "tensor apply" operation in this case becomes

$$\left[(\mathcal{T})^t \mathbf{c}^{m-1}\right]_j = \sum_{i_1,\dots,i_{m-1}=1}^{N} T_{ji_1\dots i_{m-1}}\, c_{i_1}\dots c_{i_{m-1}}. \tag{4.35}$$

However, for this operation to make sense, we need to impose uniformity in the number of head nodes of a hyperedge, so as to be able to construct a single tensor from them. Hence we arrive at the following definition.

Definition 4.22 ($\mathcal{B}$-uniformity). *Let $H = (V, E)$ be a directed hypergraph, $m \in \{1, \dots, \max_{e\in E} |e|\}$. H is said to be $\mathcal{B}$-uniform if $m = |h(e)|, \forall e \in E$.*

We will discuss and compute numerically centralities of hypergraphs fulfilling this constraint soon, but before that let us examine the other kind of hyperedges.

4.4.2.2 $\mathcal{F}$-hypergraphs

We now turn to the opposite case, that of directed hypergraphs containing only $\mathcal{F}$-hyperedges, with a fixed number of nodes per hyperedge (i.e. m-uniform)[6]. As we will see, this case is rather troublesome when it comes to centrality computations.

[6]This type is employed in [122, 97] with a specifically weighted adjacency tensor

In this case the transposition reduces to

$$(T_{ij_2\dots j_m})^t = T_{j_2\dots j_m i}. \tag{4.36}$$

If we naïvely adapt the "tensor apply" operation, we get a mixing of the indices summed over due to the transposition, in the sense that we sum over both the tail node and some head nodes.

$$\left[(\mathcal{T})^t \mathbf{c}^{m-1}\right]_{j_2} = \sum_{j_3,\dots,j_m,i=1}^{N} T_{j_2\dots j_m i}\, c_{j_3}\dots c_{j_m} c_i. \tag{4.37}$$

This is heuristically incorrect: in directed edges the centrality contribution flows from the "input" node(s) to the "output" one(s), a fact that is later reflected in the sum. However, if we take this operation at face value, we see that all head nodes will contribute to each other.

There are different alternatives to solve this issue.

1. *Projection:* We argue that the correct operation involved in the centrality calculation is not the standard "tensor apply". Instead, centrality in $\mathcal{F}$-hypergraphs should be calculated using

$$\lambda c_j = \sum_{i\to\{j,\dots,j_m\}} c_i = \frac{1}{(m-1)!} \sum_{j_3,\dots,j_m,i=1}^{N} T_{jj_3\dots j_m i}\, c_i, \tag{4.38}$$

   which is, in essence, a glorified version of the eigenvector centrality (as indeed there are $(m-1)!$ identically valued components $T_{j\sigma(j_3\dots j_m)i}$)

2. *ZEC-like:* We can invert the mathematical interpretation of the heuristics as

$$\lambda c_{j_2} c_{j_3}\dots c_{j_m} = \sum_{i=1}^{N} T_{j_2 j_3\dots j_m i}\, c_i \tag{4.39}$$

   that is, the combination (multiplication) of the centralities of the head nodes is proportional to that of the tail node pointing to them. In some way this is reminiscent of the $\mathcal{Z}$-eigenvector, and it is easy to check that, whichever the solution is (if any, as we are not sure it is guaranteed), it will not be rescalable.

3. *HEC-like:* To mitigate the rescaling issue, we can adopt a similar strategy as what the $\mathcal{H}$-eigenvector does

$$\lambda \sqrt[(m-1)]{c_{j_2} c_{j_3}\dots c_{j_m}} = \sum_{i=1}^{N} T_{j_2 j_3\dots j_m i}\, c_i \quad \text{or} \quad \lambda c_{j_2} c_{j_3}\dots c_{j_m} = \sum_{i=1}^{N} T_{j_2 j_3\dots j_m i}\, c_i^{(m-1)} \tag{4.40}$$

   This proposal, as well as the previous one, suffers from a serious conceptual problem. Suppose we have a 3-uniform $\mathcal{F}$-hypergraph $H = (V, E)$, which does not contain nodes $2, 3$ in the same head $h(e)$ of any hyperedge $e \in E$. Consider

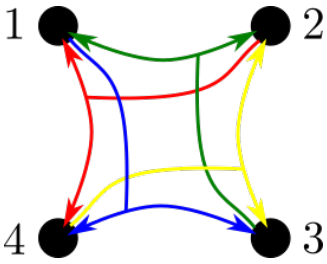
the multiplication c_2c_3: there is no transposed tensor component to sum over, so the right-hand side would be zero, hence either must be zero. However this is not true in general. So in a sense the tensor components dictate what products of centralities could be in the left-hand side of the defining equations.

4. *CEC-like:* we already put forward ZEC-like and HEC-like proposals, so why not CEC-like, where the combination of head centralities is not multiplicative but additive?

$$\lambda(c_{j_2} + c_{j_3} + \cdots + c_{j_m}) = \sum_{i=1}^{N} T_{j_2 j_3 \ldots j_m i}\, c_i \tag{4.41}$$

To discern between these alternatives, consider the following simple example.

Example 4.23. *Let $H = (V, E)$ be the hypergraph with edges*



$$E_1 = \{\{1\}, \{3,4\}\}, \quad E_2 = \{\{2\}, \{4,1\}\},$$
$$E_3 = \{\{3\}, \{1,2\}\}, \quad E_4 = \{\{4\}, \{2,3\}\}, \tag{4.42}$$

as shown in Figure 4.9.

Figure 4.9: Example hypergraph.

From the point of view of the tensor, its components are

$$T_{1\sigma(34)} = w_1, \quad T_{2\sigma(41)} = w_2, \quad T_{3\sigma(12)} = w_3, \quad T_{4\sigma(23)} = w_4. \tag{4.43}$$

Firstly, notice that H is strongly connected, as is H^t. Furthermore, note that the hypergraph is designed to be symmetrical under the transformation $i \to i + 1$ mod 4. This implies that any sensible definition of centrality for this hypergraph, if unweighted, should yield identical scores for each of the nodes.

We are now going to split the discussion between unweighted and weighted cases to see the differences between the proposals above.

1. *Unweighted ($w_1 = w_2 = w_3 = w_4 = 1$) case:*

   a) *As we discussed, the centrality is that of the directed graph, $\mathbf{c} = (1, 1, 1, 1)^T$, and by the Perron-Frobenius Theorem A.2 it is unique.*

   b) *In this case we have no guarantees of existence nor uniqueness. However, numerically we can obtain $\lambda\mathbf{c} = (1, 1, 1, 1)^T$. We would need to fix a normalization, as in the $\mathcal{Z}$-eigenvector case, which would yield a suitable eigenvalue.*

   c) *Firstly, the equations impose a somewhat strange equality, $\lambda^4 = w_1w_2w_3w_4$. In any case, we again find $\mathbf{c} = (1, 1, 1, 1)^T$.*

d) The measure fails to capture the symmetry. The result is

$$\mathbf{c} = \left(\frac{\sqrt{\sqrt{3}+1}}{2}, \frac{\sqrt{3}-1}{4}, \frac{(\sqrt{3}-1)\sqrt{\sqrt{3}+1}}{4}, \frac{(2-\sqrt{3})\sqrt{\sqrt{3}+1}}{4} \right)^T, \tag{4.44}$$

after choosing the only positive eigenvector.

2. *Weighted ($w_1 = 2$, $w_2 = w_3 = w_4 = 1$) case:*

   a) *In this case we find $\mathbf{c} = (0.93, 1.18, 1.25, 1)^T$, again unique by the Perron-Frobenius Theorem A.2 .*

   b) *Here we find $\lambda\mathbf{c} = (0.57, 0.87, 0.76, 1)^T$, not necessarily unique.*

   c) *In this case we find $\mathbf{c} = (0.54, 0.87, 0.81, 1)^T$, again not necessarily unique.*

   d) *Not worthy of discussion given the lack of symmetry in the unweighted case.*

Note that the ZEC-like and HEC-like proposals are similar, in the sense that they yield a ranking $c_4 > c_2 > c_3 > c_1$, while in the graph-like (the projection) yields $c_3 > c_2 > c_4 > c_1$. But one cold argue that the latter is more sensible, as node 4 provides centrality through a $\mathcal{F}$-hyperedge to node 3, but the converse is not true. Therefore node 1 increases both 3 and 4, and 4 increases 3, hence $c_3 > c_4$. Furthermore, the projection comes with guaranteed uniqueness and existence, which renders it the most suitable.

The fact that the most sensible operation is, as we just showed, a simple pairwise projection implies that the tensorial nature of the hypergraph is actually shadowed in this case by its pairwise directed relations. Nevertheless, there is at least a unique and easy to compute solution to this problem. And this further implies that our initial restriction to m-uniform $\mathcal{F}$-hypergraphs was not necessary: we can include hyperedges of any given size in the sum above, provided we take into account the $1/(m-1)!$ factors, with m the size of the corresponding $\mathcal{F}$-hyperedge.

4.4.2.3 General directed hypergraphs

Now that we understand how does the "tensor apply" operation act on both forward and backward hyperedges, we can discuss general directed hypergraphs and their centrality.

First, note that backward hyperedges impose a restriction on the number of nodes at the input of every interaction, namely we will require the hypergraph to be $\mathcal{B}$-uniform. On the contrary, forward hyperedges impose no restriction, as their tensor apply operation needs to be replaced by a projection-like operation.

Therefore, for a $\mathcal{B}$-uniform directed hypergraph we can define the following generalizations of the HEC centrality measures, provided it is strongly connected.

Definition 4.24 (HEC of a directed hypergraph). *Let $H = (V, E)$ be the a strongly connected, $\mathcal{B}$-uniform hypergraph, with $m_B = |h(e)|$, $\forall e \in E$ head nodes, and $m_F = \max(|t(e)|)$, $e \in E$ the maximum number of tail nodes in any hyperedge. The $\mathcal{H}$-eigenvector centrality of H is the unique, positive vector $\mathbf{c} \in \mathbb{R}^N$ satisfying*

$$\lambda c_{j_1} = \sum_{i_1,\ldots,i_{m_B}=1}^{N} \sum_{m=1}^{m_F} \frac{1}{m!} \sum_{j_1,\ldots,j_m}^{N} T_{j_1\ldots j_m i_1\ldots i_{m_B}} c_{i_1} \ldots c_{i_{m_B}}. \tag{4.45}$$

Although this definition may seem daunting, operationally it is rather clear: given a $\mathcal{B}$-uniform hypergraph where m_B is the number of head nodes (fixed due to the uniformity constraint), we turn every hyperedge into several ones, one per tail node, adjusting the combinatorial factors to avoid overcounting due to the symmetry of the adjacency tensor. We are then left with a tensor $\mathcal{T} \in \mathbb{R}^{[m_B,N]}$, whose centrality we can compute as its Perron-like eigenvector, like in the undirected case [26].

4.4.2.4 Numerical examples

Finding datasets of directed hypergraphs freely available is far from trivial, especially if we are interested in real data ones rather than synthetic ones. However, we managed to construct several of them related to chemical [123] and astro-chemical reactions[7] [117]. Once the hypergraphs have been constructed [124], we find the strongly connected sub-hypergraph satisfying the $\mathcal{B}$-uniformity condition, if any.

From these hypergraphs we can then compute their directed $\mathcal{H}$-eigenvector centrality (using the code we already developed and made available in [102]) and that of their projection graph. The top 5 nodes in each example, as obtained via the standard eigenvector centrality of the projected graph, as well as with the newly defined measure for directed hypergraphs, are shown for comparison in Table 4.5.

-	`RTG`		`unibimolecular`		`surface`		`astro`	
-	EC	HEC	EC	HEC	EC	HEC	EC	HEC
1	H	OH	H	e-	H2	H	H	E
2	OH	H	H2	Photon	CH4	CH3SH	H2	C
3	O	O	CO	O	H	CH3	CO	C+
4	CH3	CH3	C	C	CH3	H2CS	C	O
5	HCO	HCO	He	C+	NO	HNO	CN	H+

Table 4.5: First 5 elements ranked by both the eigenvector centrality (EC) of the projected graph and the directed $\mathcal{H}$-eigenvector centrality (HEC), in each of the datasets (`RTG`, `unibimolecular`, `surface` and `astro`).

We can also compare these measures in terms of the Spearman's ρ coefficient between both measures, which measures the similarity of two rankings (ranging from

[7] We also constructed others from metabolical reactions [118], but for the sake of conciseness the computation of the rankings and comparisons are left in the repository https://github.com/LaComarca-Lab/DirectedHyperCentrality.

1, meaning identical rankings, to -1, meaning completely opposite rankings). The correlation between rankings for the first K components of each hypergraph are shown in Figure 4.10.

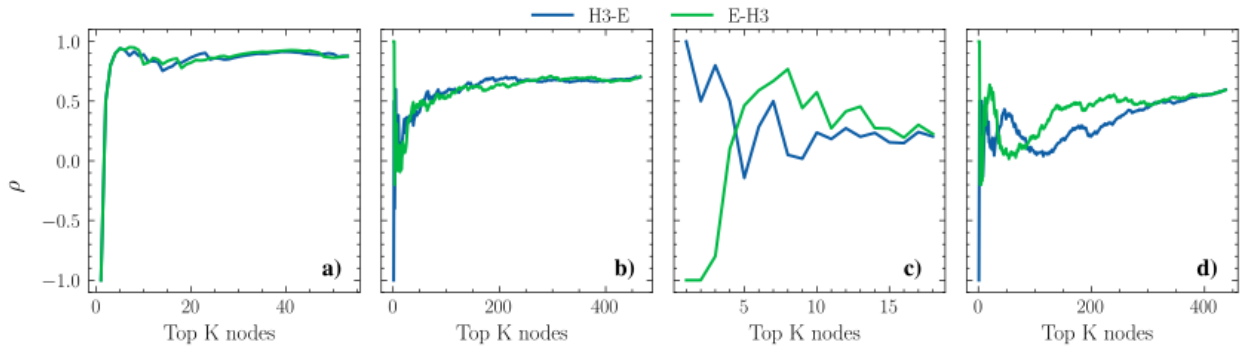


Figure 4.10: Comparison between Spearman's ρ correlation between the eigenvector centrality (E) of the projected graph and the directed $\mathcal{H}$-eigenvector centrality (H3), in the following real chemical reaction dataset hypergraphs: a) `RTG` b) `unibimolecular` c) `surface` d) `astro`.

From these figures we can see that the rankings from the pairwise projection networks and the ones from our new method are positively correlated (as they should), being very similar in some cases (e.g. subfigures **a** and **b**) but different enough that they represent a different meaning of what importance is.

4.4.3 k-step hypergraphs

In [120] the authors designed a new centrality measure (the *two-step eigenvector centrality*) for undirected graphs, which consisted of constructing a tensor (the *two-step tensor*) from powers of the adjacency matrix of the graph, which could then leverage the machinery of $\mathcal{H}$-eigenvectors in order to rank the nodes of the original graph[8].

Definition 4.25 (2-step centrality of a graph [120]). *Let $G = (V, E)$ be an undirected graph with N nodes, let $A = (a_{ij})$ be its adjacency matrix. The 2-step eigenvector centrality of the graph is the Perron-like $\mathcal{H}$-eigenvector solution to*

$$\lambda c_i = \sum_{j,k=1}^{N} T_{ijk} c_j c_k, \quad \text{where} \quad T_{ijk} = a_{ij} a_{jk}. \tag{4.46}$$

Here we will study a more general version of this method: constructing the $(k-1)$-step tensor of a directed graph and analyzing its properties.

[8]Said article is only concerned with undirected graphs and $\mathcal{H}$-eigenvectors, however there is actually no roadblock for extending it to directed graphs and $\mathcal{Z}$-eigenvectors.

Definition 4.26 (k-step hyperedge). *Let $G = (V, E)$ a possibly directed graph with adjacency matrix $A = (a_{ij})$. Consider a sequence of its nodes $i_1 \to i_2 \to \cdots \to i_k$. A k-step hyperedge corresponding to such sequence is described by the adjacency component*

$$T_{i_1 i_2 \ldots i_k} = a_{i_1 i_2} a_{i_2 i_3} \ldots a_{i_{k-1} i_k}, \tag{4.47}$$

where $T_{i_1 i_2 \ldots i_k} > 0$ iff that walk exists in the network.

It is not hard to see that a tensor constructed this way represents yet another type of heterogeneous hypergraph (one where order matters within a hyperedge but where there are no heads/tails), which we now analyze.

The most natural transposition in this case is that inherited from the transposed adjacency matrix, i.e.

$$(T_{i_1 i_2 \ldots i_{k-1} i_k})^t = a^t_{i_1 i_2} \ldots a^t_{i_{k-1} i_k} = a_{i_k i_{k-1}} \ldots a_{i_2 i_1} = T_{i_k i_{k-1} \ldots i_2 i_1}. \tag{4.48}$$

Clearly, for an undirected network we have the symmetry property $T_{i_1 i_2 \ldots i_{k-1} i_k} = T_{i_k i_{k-1} \ldots i_2 i_1}$, which is why in [120] they safely ignore the transposition step, although omitting this discussion.

We have the following, completely standard formulation of the "tensor apply" operation.

$$\left[(\mathcal{T})^t \mathbf{c}^{m-1}\right]_{i_k} = \sum_{i_{k-1} \ldots, i_1 = 1}^{N} T_{i_k i_{k-1} \ldots i_1} \, c_{i_{k-1}} \ldots c_{i_1}. \tag{4.49}$$

In some sense this measures provides centrality to the target node from its $k-1$ ancestors. This is very reminiscent of centrality measures involving the iterated line graph or the Hashimoto matrix in standard networks [71, 73].

Strong connectedness of the resulting tensor is guaranteed if the base graph is strongly connected by the following theorem.

Theorem 4.27 (Strong connectedness of the hypergraph induced by $T_{i_1 \ldots i_k}$). *Let $G = (V, E)$ be an unweighted graph with adjacency matrix A. Let H be the hypergraph induced by the tensor $T_{i_1 i_2 \ldots i_m} = a_{i_1 i_2} a_{i_2 i_3} \ldots a_{i_{k-1} i_k}$. If G is strongly connected, then H is strongly connected.*

Proof. Consider the definition of strong connectivity based on the graph G^M (see Definition A.5)

$$M_{ij} = \sum_{i_3, \ldots, i_k}^{N} T_{iji_1 \ldots i_k} = \sum_{i_3, \ldots, i_k}^{N} a_{ij} a_{ji_3} \ldots a_{i_{k-1} i_k} = a_{ij} \sum_{i_3, \ldots, i_k}^{N} a_{ji_3} \ldots a_{i_{k-1} i_k} \geq a_{ij}, \tag{4.50}$$

because if G is strongly connected then there must be at least one $(k-2)$-long walk starting from j (possibly repeating edges and nodes). Therefore $M \geq A$ component-wise, so if G is strongly connected, G^M is strongly connected and so is H. ∎

Extending the proof for positively weighted graphs is straightforward (in which case we have $M_{ij} > 0$ instead). Note that this Theorem applies to both undirected as well as directed graphs.

Numerical comparisons: To illustrate the difference between the usual eigenvector centrality and this new measure, we will compute them for three real directed networks: the Chicago road network (12982 nodes, 39018 edges), the European road network (1174 nodes, 1417 edges) and the OpenFlights dataset (2939 nodes, 30501). The choice of transport networks is due to the intrinsic relation between the k-step hypergraphs and the routing transportation application which we discussed at the end of Section 4.3.

These results again showcase the disparity in rankings with the Spearman's ρ correlation, which signals that the new measure is capturing different features (namely, it takes into account all path of length k) in the computation of the centrality scores. The results are shown in Figure 4.11.

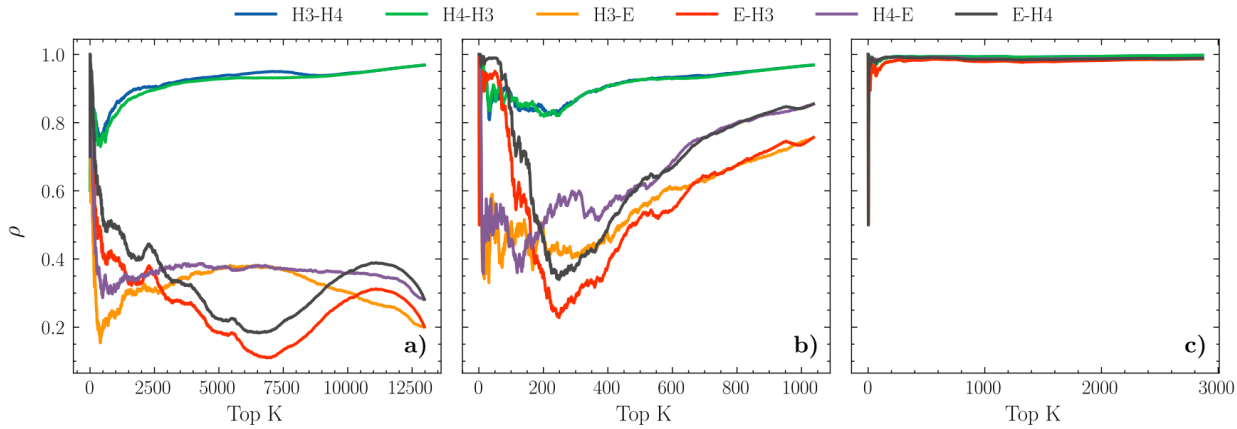


Figure 4.11: Comparison between Spearman's ρ correlation between the Eigenvector Centrality (E) of the original directed graphs and their k-step eigenvector centrality for $k = 3, 4$ (H3, H4) in the case a) Chicago roads b) European roads c) OpenFlights

In the first two subfigures one can clearly see that the correlation between the 3-step and 4-step HEC of the graph are very similar, which means that there is almost no new information when increasing the length of paths sampled with the k-step centrality. The other comparisons show a higher disparity between rankings, meaning each measure is measuring something different, being positive nonetheless.

In the last subfigure, corresponding to a flight transportation network, we find it remarkable that the rankings are almost identical, having the Spearman correlation very close to 1 for all K sampled. This last dataset corresponds to an undirected network, which, for the purposes of this articles, is treated as a directed one with both directions per edge. However this bears no relation with the high correlation, and indeed we have checked that other undirected networks (e.g. the US Power Grid) do not have this feature.

4.5 Controlling directed hypergraph centralities

In Chapter 3 we established several results regarding the controllability of the eigenvector centrality in standard networks, as well as in multilayer networks. We did not discuss how it generalized to the case of hypergraphs, essentially because we needed to firmly establish a notion of non-directedness in hypergraphs, something we discussed in Sections 4.3 and 4.4, based on our work in [125].

Something which we should keep in mind is the nonlinear nature of the HEC and ZEC measures, regardless of the hypergraph type. This implies that many results related to the controllability of the eigenvector centrality in standard graphs can't be easily extended to this new setting. Nevertheless we can make progress in one of the most central types of control throughout this manuscript, which we already discussed in the multilayer case: weight tuning. The reason for this is the fact that, even though the $\mathcal{Z}$- and $\mathcal{H}$-eigenproblems are nonlinear in the centralities, they are linear in the weights.

4.5.1 Weight tuning in undirected hypergraphs

Before moving on to directed hypergraphs, let us establish a very simple result, that of the lack of controllability of the ZEC, HEC in the case of undirected hypergraphs.

The existence and uniqueness of this centrality measure in strongly connected hypergraphs has already been assessed [97, 26]. What we may still discuss is its control. Sadly, as the following, simple counterexample shows, in general there is no control in undirected hypergraphs.

Counterexample 4.28. *Let $H = (V, E)$ be the 3-uniform hypergraph with $N = 4$ nodes and four hyperedges (all those available among its nodes). Suppose we want to obtain the centrality solution $\mathbf{c} = (0.5, 0.2, 0.1, 0.2)^T$ and for that we can assign arbitrary weights to the hyperedges*

$$T_{\sigma(123)} = w_1, \quad T_{\sigma(124)} = w_2, \quad T_{\sigma(134)} = w_3, \quad T_{\sigma(234)} = w_4, \tag{4.51}$$

where $\sigma(ijk)$ is any permutation of the three indices.

Expanding the ZEC equation (analogously for the HEC one) we reach the following linear relation

$$\lambda \begin{pmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \end{pmatrix} = 2 \begin{pmatrix} c_2c_3 & c_2c_4 & c_3c_4 & 0 \\ c_1c_3 & c_1c_4 & 0 & c_3c_4 \\ c_1c_2 & 0 & c_1c_4 & c_2c_4 \\ 0 & c_1c_2 & c_1c_3 & c_2c_3 \end{pmatrix} \cdot \begin{pmatrix} w_1 \\ w_2 \\ w_3 \\ w_4 \end{pmatrix}. \tag{4.52}$$

Solving for the necessary weights leads to

$$w_1 = 7.33333\lambda, \quad w_2 = 5.16667\lambda, \quad w_3 = 7.33333\lambda, \quad w_4 = -34.1667\lambda, \tag{4.53}$$

but that requires negative weights, therefore there is no possible full control solution in this setting.

The problem is the same as the one with standard graphs: if hyperedges are undirected their contribution to each node is the same.

A similar conclusion is found for the HEC controllability.

4.5.2 Weight tuning in heterogeneous hypergraphs

In heterogeneous hypergraphs we can hope to achieve some control of the centralities upon weight tuning, as we did in standard graphs when we considered directed, strongly connected ones. However, the space of non-undirected hypergraphs is vastly different from that of graphs, as we have stressed on many occasions throughout this Chapter.

In particular, we will see that, out of the three different heterogeneous hypergraph types defined in Section 4, the only one in which weight tuning can actually be leveraged to control the centrality scores is that of directed hypergraphs.

4.5.2.1 Cyclical hypergraphs

As advanced, there will be no control in this type of hypergraph when we tune its centralities. The reason for this is rather simple: already in Subsection 4.4.1 when we introduced the centralities of such structures, we noticed the parallel between them and those of their undirected counterparts. This turns out to eliminate the chance of weight tuning control in most networks.

Counterexample 4.29.

We consider the hypergraph with $N = 4$ nodes and cyclic edges

$$T_{\sigma_{\text{even}}(123)} = w_1, \quad T_{\sigma_{\text{even}}(234)} = w_2, \quad T_{\sigma_{\text{even}}(341)} = w_3, \quad T_{\sigma_{\text{even}}(412)} = w_4. \tag{4.54}$$

We get the exact same equation as in the undirected case, up to a factor of 2, as we no longer sum twice each tensor component (each "orientation" is separate).

Once again, we can trace this impossibility to the fact that the cyclic hyperedge weight still affects every node of the hyperedge equally.

It is clear that we need asymmetry in the hyperedges, as it happened in the directed graph case, in order to allow for some sort of control.

4.5.2.2 Directed hypergraphs

Let us start with $\mathcal{B}$-hypergraphs. Although at first glance it might seem that the tensorial nature of the problem allows for little analytical progress, the key idea is that the problem is linear in the weights (in the tensor components). Hence, a similar reasoning as the one of the original work [24] can be applied here.

Theorem 4.30 (Full control of $\mathcal{B}$-hypergraph centralities). *Let $H = (V, E)$ be a strongly connected, directed m-uniform $\mathcal{B}$-hypergraph. Let $\mathbf{c} \in \mathbb{R}^N$ be a positive*

vector. We can always choose a positive weighting $W : E \to \mathbb{R}^+$ *such that* $\mathbf{c}$ *is the ZEC or HEC of* H*.*

Proof. Let $N = |V|$, $L = |E|$. Without loss of generality we will choose the HEC case. There, we can rewrite the $\mathcal{H}$-eigenvector equation as

$$B\mathbf{w} = \lambda \mathbf{c} \tag{4.55}$$

where $\mathbf{w} \in \mathbb{R}^L$, and $B \in \mathbb{R}^{N \times L}$ is a rectangular matrix containing combinations of $m-1$ centrality scores $c_i \in \mathbf{c}$. This system has an infinite number of solutions, as strong connectedness requires $L > N$ weights and $\text{rank}(B) = N < L$ (all the equations are separated and involving each of the weights just once). □

We can actually construct an explicit solution for the weights (not necessarily unique) as follows. We uniquely identify each weight as $w_{(j,\ell)}$, where (j,ℓ), $\ell = 1, \dots k_j^{\text{in}}$ are all the hyperarcs whose head is node j (there are k_j^{in} of them), and with source nodes $\{i_1^\ell, i_2^\ell, \dots, i_{m-1}^\ell\}$. Then, we can write the previous equation as

$$\sum_{l=1}^{k_j^{\text{in}}} w_{(j,\ell)} c_{i_1^\ell} c_{i_2^\ell} \dots c_{i_{m-1}^\ell} = \lambda c_j^2. \tag{4.56}$$

A solution can now be computed directly as

$$w_{(j,1)} = w_{(j,1)} = \dots = w_{(j,k_j^{\text{in}})} = \frac{\lambda c_j^2}{\sum_{l=1}^{k_j^{\text{in}}} c_{i_1^\ell} c_{i_2^\ell} \dots c_{i_{m-1}^\ell}}. \tag{4.57}$$

Example 4.31. *Consider a hypergraph* $H = (V, E)$ *with* 4 *nodes and positively weighted hyperedges corresponding to tensor components*

$$T_{123} = T_{213} = w_1, \quad T_{142} = T_{412} = w_2, \quad T_{341} = T_{431} = w_3, \quad T_{234} = T_{324} = w_4. \tag{4.58}$$

It is easy to check that both the hypergraph and its transpose are strongly connected. In this case we can fully control the centrality of the hypergraph

$$\mathcal{Z}: \quad (w_1, w_2, w_3, w_4) = \frac{\lambda}{2}\left(\frac{c_3}{c_1 c_2}, \frac{c_2}{c_1 c_4}, \frac{c_1}{c_3 c_4}, \frac{c_4}{c_2 c_3}\right), \tag{4.59}$$

$$\mathcal{H}: \quad (w_1, w_2, w_3, w_4) = \frac{\lambda}{2}\left(\frac{c_3^2}{c_1 c_2}, \frac{c_2^2}{c_1 c_4}, \frac{c_1^2}{c_3 c_4}, \frac{c_4^2}{c_2 c_3}\right). \tag{4.60}$$

The case for $\mathcal{F}$-hypergraphs is even simpler: given that the most sensible way of computing its associated HEC centrality is via the projection of the hyperedges into pairwise components, the problem reduces to that of directed graphs, solved in [24].

Finally, note that for general directed hypergraphs, as long as they are $\mathcal{B}$-uniform (see Definition 4.22) the problem is still linear in the weights, and given by Definition 4.24. Hence a similar argument as in Theorem 4.30 holds in this case, and it enables the following result:

Theorem 4.32. *Let $H = (V, E)$ be a strongly connected, directed $\mathcal{B}$-uniform hypergraph. Let $\mathbf{c} \in \mathbb{R}^N$ be a positive vector. We can always choose a positive weighting $W : E \to \mathbb{R}^+$ such that $\mathbf{c}$ is the ZEC or HEC of H.*

4.5.2.3 k-step hypergraphs

Lastly, we want to consider the generalization of the 2-step hypergraph defined in [120] to directed k-steps, as examined in the previous Section. Given that these heterogeneous hypergraph variants arise from paths in a standard graph, where the multiplication of the adjacency matrix' components along the path constitutes the hyperedge weight, we assume that the controllability may come from tuning these adjacency matrix weights.

It may come as a surprise that there is still no control in this case. To see it, consider the following counterexample.

Counterexample 4.33. *Consider the directed graph G consisting of two nodes joined by two directed edges. This graph is clearly strongly connected. The associated two-step tensor only has two components, $T_{121} = a_{12}a_{21} = T_{212}$. The associated equations are*

$$\begin{cases} \lambda c_1^2 &= T_{121} c_1 c_2 \\ \lambda c_2^2 &= T_{212} c_2 c_1 \end{cases} \Rightarrow \begin{cases} \lambda c_1^2 &= a_{12}a_{21}\, c_1 c_2 \\ \lambda c_2^2 &= a_{21}a_{12}\, c_2 c_1 \end{cases} \Rightarrow c_1^2 = c_2^2 \Rightarrow c_1 = c_2, \tag{4.61}$$

where the last step is due to positivity of the centrality scores. Therefore, whatever the value of both weights, the centralities will be exactly identical.

It could seem that this example is particular because although there are two tensor components, in terms of the adjacency elements, which are the controllable knobs, they are not independent. However, this is still uncontrollable in less simple graphs. For instance, in a graph G with directed edges $1 \leftrightarrow 2, 1 \to 3, 3 \to 2$ one would need to set a_{32} to zero in order to obtain $\mathbf{c} = (1/4, 1/4, 1/2)^T$ or $\mathbf{c} = (1/3, 1/3, 1/3)^T$.

Chapter 5
Conclusions and outlook

It's been more than 20 years since the explosion of popularity of network science, but the scientific interest in exploring those research avenues has not faded ever since, having instead transitioned from standard networks to more complex mathematical abstractions, such as multilayer networks and hypergraphs. Furthermore, centrality measures is still one of the flagships of complex network theory, with an active community which puts forward both theoretical advances and applies them to real applications. Theoretical advances usually come in two varieties: either the proposal of new, sensible centrality definitions (based on different heuristics or applicable to different types of network structures) or novel analysis of already established ones, providing an enhanced understanding of their properties. As for real applications, they tend to require a deep understanding of the system in question, to distill the key features in them and the heuristics they shall satisfy.

In this thesis we have focused on the former, always striving towards a deeper understanding of the behavior and expected outcomes of some of the most significant measures, those of a spectral nature. In particular:

- We provided a comprehensive overview of the possibilities of controlling several of these centrality measures; we analytically exploited a plethora techniques, ranging from weight tuning and adding self-loops to modifying the measure's parameters, understanding the extent to which measures could be altered to obtain certain outcomes. The exploration has been far from exhaustive, for there are too many possibilities: throughout this manuscript we have discussed around a dozen of centrality measures for graphs, multilayer networks and hypergraphs; and several control scenarios as those we just mentioned. Despite this, hopefully we have been able to convey a sense of cohesion and interest in the subjects discussed.

- We put forward a series of extensions and generalizations of spectral centralities in hypergraphs, exposing the shortcomings of already established methods (namely, the lack of applicability to non-uniform and directed/heterogeneous hypergraphs) to then offer analytically consistent and numerically sound alternatives based on the Perron-Frobenius theory for tensors. We then reintroduced

the weight tuning control paradigm in these proposals to assess their controllability.

As we mentioned in the beginning, the research lines which were pursued during the completion of this thesis are mostly theoretical, focused on understanding rather than applying. As such, they might serve as testing grounds for new measures and proposals, assessing their controllability as soon as they are designed. Moreover, some of the methods which have been developed to study these topics are of interest in more generality, as they mostly make references to spectral properties of mathematical objects, rather than to specific networks.

However, one may wonder whether these are just mathematical musings or if, on the contrary, there could potentially be applications in real world systems. This is a rather complicated question, and while we have given glimpses of answers throughout this thesis, answering it in all generality would require establishing new research lines, counting with the knowledge of experts in each of the areas involved, something which is out of the present scope. Nevertheless, it is a line of inquiry worth exploring, which will surely shed light on emergent properties of complex systems of various types.

Moving forward, we see several directions for potential research; here we highlight those which, in our opinion, are the most interesting and challenging.

- As discussed above, understanding the consequences of the obtained bounds, results, and control methods in real systems is of utmost importance. The original goal of centrality measures, that is, identifying key individuals in complex systems, is relevant to this day. Moreover, the rapid evolution of technology and society requires advancements in these identification techniques, as well as an in-depth analysis of their possibilities and limitations, as in this thesis.

- The combination of control paradigms and centrality measures is too vast; the list of inspected cases was definitely not exhaustive. In fact, we did not introduce other well-known spectral centrality measures, such as the Hubbell index [126] or the HITS algorithm [127], because of the lack of representation in this work.

  Regarding structural changes, they were mostly focused on the eigenvector centrality, for its analytical tractability and general interest. It would be interesting to investigate whether these structural changes allow for some form of control in other spectral measures.

  However, in the case of parametric control, we focused our attention on PageRank and some of its variants (biplex and node-dependent restart). These are, by far, not the only parametric spectral centrality measures; therefore, it will be worth exploring whether similar techniques hold for others.

  All of these questions are clear pathways through which progress can be achieved, supplementing, and possibly generalizing the results obtained in this thesis.

- Lastly, everything we have discussed has referred to structural properties of the networks: in the end, centrality measures are just that. While this is sufficient for many applications, there are many others that also require taking dynamical processes occurring on the network into account (e.g., synchronization, epidemic spreading, percolation, etc.). Their phenomenology thoroughly depends on the underlying network; hence, it remains to understand the relationship between the studied controllability properties (and the key individuals that are then obtained) and their effects on these dynamical processes.

Appendix A
The Perron-Frobenius Theorem

One of the most important results which is used throughout this thesis is the Perron-Frobenius Theorem [61, 62]. This is a fundamental result in linear algebra, which is often overlooked in undergraduate courses, usually relegated to the one of the last chapters of their textbooks [33]. However, it has very important practical applications: on the one hand it concerns the spectral radius of matrices, and its associated eigenvector, both of which are very important in numerical methods and for steady states of Markov chains, to mention a few examples. On the other hand, it applies to non negative matrices, which are ubiquitous in many situations such as the ones described above, as well as many others [128].

In Section A.1 we will present the Perron-Frobenius Theorem and its main ingredient, which is matrix irreducibility. We then discuss its consequences in graph theory, for future reference throughout the main text. Later in Section A.2 we will present a generalization of these ideas and results (tensor irreducibility and a generalized Perron-Frobenius Theorem for certain tensor eigenproblems), which will also be referenced several times throughout Chapter 4.

A.1 Irreducibility in graphs

Let us begin with an important definition, that of matrix reducibility.

Definition A.1 (Reducible matrix). *A matrix A is said to be reducible if there exists a permutation matrix P such that*

$$P^T AP = \begin{pmatrix} B & C \\ 0 & D \end{pmatrix} \tag{A.1}$$

where B and C are both square matrices. Otherwise A is said to be irreducible.

Note that if a matrix is irreducible, so is its transpose.

Let us think for a moment what this means for the adjacency matrix of a directed graph. Firstly, from the perspective of the graph a permutation $P^T AP$ is simply a relabeling of the nodes (as it is just an interchange of the same rows as columns).

Thus, if an adjacency matrix is reducible, then its associated graph is isomorphic (under simple relabeling) to a graph which is not strongly connected, as the subgraph associated to the submatrix B points to the subgraph associated to the submatrix D but not the converse.

This connection between algebra and graph theory is so important that the definition of an irreducible matrix is often alternatively stated as a matrix whose associated directed graph is strongly connected.

Recall now that for a square matrix A its spectrum (set of eigenvalues) is denoted as $\sigma(A)$ and its spectral radius as $\rho(A) = \max_{\lambda_i \in \sigma(A)}\{|\lambda_i|\}$. We can finally state the following foundational result.

Theorem A.2 (Perron-Frobenius [61, 62]). *Let $A \in \mathbb{R}^{N\times N}$ be a non-negative, irreducible square matrix. Then the following statements are true:*

1. *$\rho(A) \in \sigma(A)$ and its algebraic multiplicity is 1.*
2. *There exists a vector $\mathbf{c} \in \mathbb{R}^n$ with $\mathbf{c} > 0$ such that it is an eigenvector of A associated to the eigenvalue $\rho(A)$.*
3. *The eigenvector $\mathbf{c}$ is unique up to scaling.*

The eigenvector $\mathbf{c}$ is sometimes referred to as the Perron eigenvector of A.

It is important to note that there is also the preceding Perron theorem [61], which is analogous and applies to positive square matrices without the need to consider irreducibility. This is relevant in the case of PageRank (where the addition of the personalization part renders the Google matrix positive).

To summarize, we now know that any strongly connected graph has an irreducible adjacency matrix, which in turn implies that there is a unique positive eigenvector associated to its spectral radius.

Before moving on, we should remark that there has been an implicit assumption in this discussion, which is the fact that we only consider positively weighted graphs, to fulfill the non-negativity requirement. Even though there is a plethora of systems described with positive weights, there are some works extending this Theorem to matrices with a few, localized, negative entries [129, 130].

A.2 Irreducible tensors and hypergraphs

Similarly to the matricial case, we can distinguish between reducible and irreducible tensors, but with a bit of a twist.

Definition A.3 (Irreducible tensor [97]). *An order-m, dimension-N tensor $T \in \mathbb{R}^{[m,N]}$ is reducible if there is a nonempty proper index subset $J \subset \{1, \dots, N\}$ such that*

$$T_{i_1 \dots i_m} = 0,\ \forall i_1 \in J,\ \forall i_2, \dots, i_m \notin J. \tag{A.2}$$

If T is not reducible, then it is irreducible.

A number of results have been proven relating connectedness properties of hypergraphs to this the irreducibility of this tensor [131, 97]. However, unlike the pairwise case, the intuitive notion of connectedness in a hypergraph does not directly translate to irreducibility of the associated tensor. We can see this in the following example [97].

Example A.4. *Consider the following undirected hypergraph $H = (V, E)$, with $V = \{1, 2, 3, 4\}$ and $E = \{(1, 2, 3), (2, 3, 4)\}$. Its adjacency tensor $\mathcal{T} = (T_{ijk}) \in \mathbb{R}^{[3,4]}$ has components*

$$T_{\sigma(123)} = T_{\sigma(234)} = 1, \quad T_{ijk} = 0 \quad \text{otherwise.} \tag{A.3}$$

where $\sigma(ijk)$ indicates any permutation of i, j, k. Let $J = \{2, 3\}$. Then J is a nonempty proper subset of $\{1, 2, 3, 4\}$. We see that $a_{ijk} \equiv 0$ for all $i \in J$ and $j, k \notin J$. Thus, $\mathcal{T}$ is reducible. However, it is clear that the associated hypergraph is strongly connected.

As it happens, tensor irreducibility is too strong a constraint to fully describe general hypergraphs. Instead, strongly connected hypergraphs are described in terms of *weakly* irreducible tensors.

Definition A.5 (Weakly irreducible tensor [97]). *Let $M = (M_{ij})$ be a $N \times N$ non-negative matrix defined by*

$$M_{ij} = \sum_{j_3, \dots, j_m = 1}^{N} |T_{ij j_3 \dots j_m}|. \tag{A.4}$$

Then $\mathcal{T}$ is weakly irreducible if and only if M is irreducible.

A hypergraph is said to be strongly connected if its adjacency tensor is irreducible in the previous sense, and we will often denote as G^M the graph whose adjacency matrix is M. This definition is equivalent to the intuitive notion of connectedness, as G^M will have an edge between nodes i, j if there is at least a hyperedge containing them. Luckily, most of the existence results which will be relevant for us have also been proven for these types of tensors [97].

We can now state the following result.

Theorem A.6 (Generalized Perron-Frobenius Theorem for the weakly irreducible tensors [97]). *. Let $T \in \mathbb{R}^{m,N}$ be a weakly irreducible tensor. Then there exists*

- *$\mathcal{Z}$-eigenpair $(\lambda, \mathbf{c})$, $\lambda > 0, \mathbf{c} > 0$ satisfying $\lambda \mathbf{c} = \mathcal{T}\mathbf{c}^{m-1}$, $||\mathbf{c}||_1 = 1$.*
- *$\mathcal{H}$-eigenpair $(\lambda, \mathbf{c})$, $\lambda > 0, \mathbf{c} > 0$ satisfying $\lambda \mathbf{c}^{m-1} = \mathcal{T}\mathbf{c}^{m-1}$ where λ is the largest $\mathcal{H}$-eigenvalue of $\mathcal{T}$ and $\mathbf{c}$ is unique up to scaling.*

Appendix B

Uplift in $\mathcal{Z}$-eigenvectors

As we discussed in Section 4.2, the ZEC centrality can't be computed when uplifting a non-uniform hypergraph, as the different sums (one per order of interaction) can only be grouped together if we are able to rescale the centrality such that $c_\star = 1$, which we can't if we are considering $\mathcal{Z}$-eigenvectors. However, this is not an issue if we start from a 2-uniform hypergraph (in other words, a pairwise graph).

In fact, $\mathcal{Z}$-eigenvectors allow us to generalize the uplift operation to having more than one different auxiliary node[1], e.g. $\star_1, ..., \star_k$.

Definition B.1 (Multi-Uplifted hypergraph). *Let $H = (V, E)$ be an M-uniform hypergraph and let $m \geq M$. We define the multi-uplifted hypergraph at order m with s auxiliary nodes, each contained p_k times within each hyperedge as*

$$\widetilde{H} = (\widetilde{V}, \widetilde{E}), \quad \textit{where} \quad \widetilde{V} = V \cup \{\star_1, ..., \star_s\} \quad \textit{and} \quad \widetilde{E} = \left\{ e \cup \left(\bigcup_{k=1}^{s} \bigcup_{l=1}^{p_k} \{\star_k\} \right), e \in E \right\}, \tag{B.1}$$

with $\sum_{k=1}^{s} p_k = m - M$.

As we previously declared, this operation on 2-uniform (standard) graphs allow us to relate the $\mathcal{Z}$-eigenvectors of the adjacency tensor of hypergraphs to those of their original, standard graph. To see this, consider adding two different auxiliary nodes $\star, \times$ to a graph G with adjacency matrix $A = (A_{ij})$. This operation translates into the following rewriting:

$$\sum_{j=1}^{N} A_{ij} c_j \to \sum_{j,k,l=1}^{N,\star,\times} \widetilde{T}_{ijkl} c_j \, c_k \, c_l = \binom{3}{2} \sum_{k,l=1}^{N} \widetilde{T}_{ij\star\times} c_j c_\star \, c_\times, \tag{B.2}$$

where the notation $\sum_{i=1}^{N,\star,\times}$ indicates summing over the index $i \in \{1, ..., N\} \cup \{\star, \times\}$.

[1] This generalization was already possible in the HEC case, however in that case it only cluttered the notation and hampered the calculation, as the computational complexity scales with the number of *distinct* nodes involved. Note also that in that case further conditions would be required for a well-defined uplift, as in order to be able to scale the centrality such that $c_{\star_k} = 1 \, \forall k$, we need all of them to be indistiguishable from each other, i.e. they must be related by permutation.

Given that, by definition, $\widetilde{T}_{ij\star\times} = \frac{1}{12}A_{ij}$, the $\mathcal{Z}$-eigenvector equation of the uplifted 4-uniform hypergraph is equivalent to the $\mathcal{Z}$-eigenvector equation of the original 2-uniform hypergraph, which reduces to the standard eigenvector centrality of the graph:

$$\lambda \mathbf{c} = \widetilde{T}\mathbf{c}^3, \quad \mathbf{c} = (c_1, ..., c_N, c_\star, c_\times)^T \Longleftrightarrow \lambda' \mathbf{c}' = A(\mathbf{c}')^2, \quad \begin{cases} \lambda' & = \dfrac{4\lambda}{c_\star c_\times}, \\ \mathbf{c}' & = (c_1, ..., c_N)^T \end{cases}. \tag{B.3}$$

We can extrapolate this example to the uplift from a 2-uniform hypergraph to an $(2+l)$-uniform hypergraph, as stated in the following theorem.

Theorem B.2 (Correspondence between $\mathcal{Z}$-eigenvectors). *Let $\widetilde{H} = (\widetilde{V}, \widetilde{E})$ be a strongly connected, $(2+l)$-uniform hypergraph with $l \geq 1$. If there is a non-empty subset of nodes $V^\star = \{\star_1, ...\star_s\} \subset \widetilde{V}$, each contained $\{p_1, ..., p_s\}$ times, respectively, in every hyperedge, such that $\sum_{i=1}^s p_i = l$, then,*

- *The components of the $\mathcal{Z}$-eigenvectors of $\widetilde{H}$ associated to the nodes $i \in V = \widetilde{V} \backslash V^\star$ correspond to those of the 2-uniform hypergraph $H = (V, E)$ having those s nodes removed.*
- *The components of the positive $\mathcal{Z}$-eigenvectors of $\widetilde{H}$ associated to the auxiliary nodes $n \in V^\star$ are uniquely determined by the other components.*
- *The $\mathcal{Z}$-eigenvalues $\widetilde{\lambda}$ of $\widetilde{H}$ correspond to those of the 2-uniform hypergraph H, λ, rescaled as*

$$\widetilde{\lambda} = \lambda \Omega (\widetilde{c}_{\star_1})^{p_1} ... (\widetilde{c}_{\star_s})^{p_s}, \quad \Omega = \frac{(l+1)!}{\prod_{i=1}^s p_i!}. \tag{B.4}$$

Proof. Under the conditions stated, $\widetilde{H}$ can be viewed as an uplift of the hypergraph H with s auxiliary nodes, each one contained equally in each and every hyperedge. The $\mathcal{Z}$-eigenvector equation for the uplifted hypergraph can be written as

$$\widetilde{\lambda}\widetilde{c}_{i_1} = \sum_{i_2,...,i_{2+l}=1}^{N,\star_1,...,\star_s} \widetilde{T}_{i_1,...,i_{2+l}} \widetilde{c}_{i_2} ... \widetilde{c}_{i_{2+l}} = \Omega \sum_{i_2=1}^{N} T_{i_1,i_2} \widetilde{c}_{i_2} (\widetilde{c}_{\star_1})^{p_1} ... (\widetilde{c}_{\star_s})^{p_s}, \tag{B.5}$$

where we have summed over the auxiliary nodes, recovering the pre-uplifted tensor T components times a combinatorial factor Ω, product of the symmetry of the adjacency tensor. We now carefully calculate this factor.

1. In the equation for node i, there will be a sum over $2 + l - 1 = l + 1$ indices (1 corresponds to its real j-th neighbor, l to the auxiliary nodes added). We will have all possible $(l+1)!$ permutations.
2. We need to subtract the repetitions of auxiliary nodes, given by their multiplicities p_i.

Having both of these facts considered we can easily calculate it to be

$$\Omega = \frac{(l+1)!}{\prod_{i=1}^{s} p_i!}. \tag{B.6}$$

The centralities of these auxiliary nodes can now be pulled out of the sum, obtaining

$$\lambda \widetilde{c}_{i_1} = \sum_{i_2=1}^{N} T_{i_1,i_2} \widetilde{c}_{i_2}, \quad \widetilde{\lambda} = \lambda \Omega (\widetilde{c}_{\star_1})^{p_1} ... (\widetilde{c}_{\star_s})^{p_s}, \tag{B.7}$$

where we have already rescaled the $\mathcal{Z}$-eigenvalue accordingly. Noticing that $T = A$ with A being the adjacency matrix of G, we arrive at the equation $\lambda \mathbf{c} = A\mathbf{c}$, which is precisely the eigenvector equation of the 2-uniform hypergraph (pairwise graph) G.

Therefore, the first N components of the $\mathcal{Z}$-eigenvector of the uplifted hypergraph $\widetilde{H}$ correspond to the eigenvector $\mathbf{c}$ associated to G.

It is left for us to discuss the behavior of the remaining equations, one per auxiliary node. Without loss of generality, we consider the equation of node $\star_1$:

$$\widetilde{\lambda} \widetilde{c}_{\star_1} = \sum_{i_2,...,i_{2+l}=1}^{N,\star_1,...,\star_s} \widetilde{T}_{\star_1,...,i_{2+l}} \, \widetilde{c}_{i_2} \, ... \, \widetilde{c}_{i_{2+l}} = \Omega \sum_{i_1,i_2=1}^{N} T_{i_1,i_2} \, \widetilde{c}_{i_1} \, \widetilde{c}_{i_2} \, (\widetilde{c}_{\star_1})^{p_1-1} ... \, (\widetilde{c}_{\star_s})^{p_s}. \tag{B.8}$$

Multiplying both sides by $c_{\star_1}$ and then replacing $\widetilde{\lambda}$ in term of λ, we obtain the following expression

$$\lambda (\widetilde{c}_{\star_1})^2 = \sum_{i_1,i_2=1}^{N} T_{i_1,i_2} \widetilde{c}_{i_1} \, \widetilde{c}_{i_2}. \tag{B.9}$$

Therefore, each component associated to an auxiliary node is uniquely determined (up to a sign, although we can always choose the positive solution) by the components of the non-auxiliary nodes. □

Remark B.3. *We have omitted the norm constraint required for $\mathcal{Z}_1$ or $\mathcal{Z}_2$-eigenvectors. We are allowed to do so because we uplift a pairwise graph: eigenvectors of the adjacency matrix can be rescaled as will, therefore the first n components of the $\mathcal{Z}$-eigenvector of the uplifted hypergraph $\widetilde{H}$ can be matched to a specific scaling of the eigenvector of the adjacency matrix of G.*

Note that this is the reason why this theorem can't be generalized to an uplift from an m-uniform hypergraph to an $(m+l)$-uniform hypergraph: even though the $\mathcal{Z}$-eigenvector equations can be related to each other, in general their norm constraints will be incompatible.

For illustrative purposes we provide an example which can be analytically solved, following the uplift on Figure B.1.

Example B.4. *Consider the graph $G = (V, E)$ with nodeset $V = \{1, 2, 3\}$ and edgeset $E = \{\{1, 2\}, \{2, 3\}\}$). It can be seen as a 2-uniform hypergraph $H \simeq G$.*

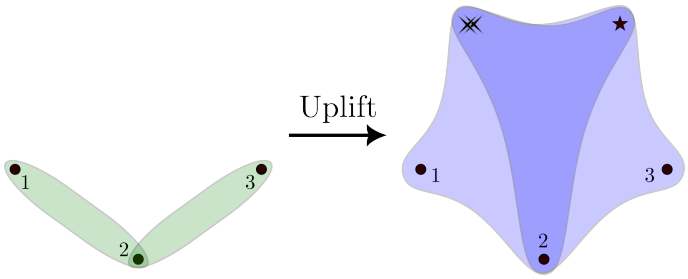


Figure B.1: Uplift of a 2-uniform hypergraph (a pairwise graph) to 5-uniform by adding two nodes $\star, \times$ to each hyperedge, the latter being added twice (depicted as there being two of them for illustrative purposes).

Suppose we uplift it to a 5-uniform hypergraph $\widetilde{H}$, adding auxiliary nodes $\star$ and $\times$ in each hyperedge, the former once and the latter two times, i.e.

$$\widetilde{H} = (\widetilde{V}, \widetilde{E}), \quad \widetilde{V} = \{1, 2, 3, \star, \times\}, \quad \widetilde{E} = \{\{1, 2, \star, \times, \times\}, \{2, 3, \star, \times, \times\}\}, \tag{B.10}$$

as shown in Figure B.1.

The first thing we would need to do is rescaling the adjacency matrix into the hypergraph tensor with suitable combinatorial factors (as in Definition 4.8). However, here we can omit this step, as this factor is the same for all components. This implies that it will only modify the $\mathcal{Z}$-eigenvalue, but that is something we will already compute.

The $\mathcal{Z}$-eigenvector equation of $\widetilde{H}$ decouples into three distinct ones:

- *Three equations for the centrality of the original nodes ($i \in \{1, 2, 3\}$),*

$$\lambda c_i = \sum_{j,k,l,m} T_{ijklm} c_j\, c_k\, c_l\, c_m = \frac{4!}{2!} \sum_{j=1}^{N} T_{ij\star\times\times} c_j\, c_\star\, c_\times^2 = 12 c_\star\, c_\times^2 \sum_{j=1}^{N} A_{ij} c_j, \tag{B.11}$$

  where this combinatorial factor is the product of fixing 4 indices, out of which 2 are repeated.

- *An equation for the centrality of the auxiliary node $\star$.*

$$\begin{aligned} \lambda c_\star &= \sum_{j,k,l,m} T_{\star jklm} c_j c_k c_l c_m = \frac{4!}{2!} (T_{\star 12\times\times} c_1 + T_{\star 23\times\times} c_3)\, c_2\, c_\times^2 \\ &= 12 (A_{12} c_1 + A_{23} c_3)\, c_2\, c_\times^2. \end{aligned} \tag{B.12}$$

- *An equation for the centrality of the auxiliary node $\times$.*

$$\begin{aligned} \lambda c_\times &= \sum_{j,k,l,m} T_{\times jklm} c_j c_k c_l c_m = 4! (T_{\times 12\star\times} c_1 + T_{\times 23\star\times} c_3)\, c_2\, c_\star\, c_\times \\ &= 24 (A_{12} c_1 + A_{23} c_3)\, c_2\, c_\star\, c_\times. \end{aligned} \tag{B.13}$$

If rescale $\lambda' = \lambda/(12c_\star c_\times^2)$, we have that the first of them becomes $\lambda'\mathbf{c} = A\mathbf{c}$; in other words, it is the eigenvector equation of the adjacency matrix of the original graph G. As it is connected, we are guaranteed to have a unique, positive solution $\mathbf{c} > 0$.

The remaining equations are then (almost) completely fixed, as after re-scaling the eigenvalue leads to

$$\lambda' c_\star^2 = (A_{12}c_1 + A_{23}c_3)\, c_2, \quad \lambda' c_\times^2 = 2(A_{12}c_1 + A_{23}c_3)\, c_2. \tag{B.14}$$

which not only enforces $\sqrt{2}c_\star = c_\times$ but also guarantees their positivity, as $A_{12} = A_{23}$ and $\mathbf{c} > 0$.

There is yet a subtlety to take into account: even though the Perron eigenvector $\mathbf{c}$ can be rescaled as $\mathbf{c}' = \alpha\mathbf{c}$, the $\mathcal{Z}$-eigenvector including $c_\star, c_\times$ cannot, it requires some normalization ($\mathbf{c}^T\mathbf{c} = 1$ or $||\mathbf{c}||_1 = 1$), which will force upon your solution the suitable value of α.

With all these taken into account, we find the unique, positive solution $\tilde{c} = (\mathbf{c}, c_\star, c_\times)^T$ to the problem to be

$$\mathcal{Z}_1 : \tilde{c} = \frac{1}{4+2\sqrt{2}}\left(1, \sqrt{2}, 1, \sqrt{2}, 2\right)^T; \quad \mathcal{Z}_2 : \tilde{c} = \frac{1}{5}\left(1, \sqrt{2}, 1, \sqrt{2}, 2\right)^T. \tag{B.15}$$

We can finally obtain the following sufficient condition for existence and uniqueness of certain $\mathcal{Z}$-eigenvectors of tensors.

Corollary B.5 (Sufficient condition for the existence of the Perron-like $\mathcal{Z}$-eigenvector). *Let T be a symmetric tensor of order $m > 2$. If its associated hypergraph H is strongly connected and can be seen as an uplift from a pairwise graph G, then a Perron-like $\mathcal{Z}$-eigenvector of T (i.e. a unique, positive $\mathcal{Z}$-eigenvector) is guaranteed to exist.*

Proof. This follows directly from the Perron-Frobenius theorem, as it guarantees the existence and uniqueness of the eigenvector $\mathbf{c}$ of the graph G and the fact that the remaining (auxiliary nodes) equations fix uniquely (after choosing their positive values) these components in terms of $\mathbf{c}$. □

The only thing left for us to discuss is the connection between this multilinear algebra result, and our original perspective, which was that of hypergraph centralities. But making this leap is rather evident.

Corollary B.6 (Sufficient condition for the uniqueness of ZEC). *If a hypergraph H is strongly connected and can be seen as an uplift from a pairwise graph G, then it has a unique Perron-like $\mathcal{Z}$-eigenvector.*

It is important to remark that the computation of $\mathcal{Z}$-eigenvalues is particularly complicated (See e.g. [132, 26, 133]), however our work clears the path for the simple computation of a whole class of hypergraphs.

Bibliography

[1] Klaus Lehnertz. "Synchronization phenomena in human epileptic brain networks." In: *Journal of neuroscience methods.* 183.1 (2009-09).

[2] Amy N. Langville and Carl D. Meyer. *Google's PageRank and beyond, the science of search engine rankings.* Princeton Univ. Press, 2006.

[3] Steven H Strogatz. "Human sleep and circadian rhythms: a simple model based on two coupled oscillators". In: *Journal of Mathematical Biology* 25.3 (1987), pp. 327–347.

[4] Romualdo Pastor-Satorras, Claudio Castellano, Piet Van Mieghem, and Alessandro Vespignani. "Epidemic processes in complex networks". In: *Reviews of modern physics* 87.3 (2015), p. 925.

[5] Andrzej Grabowski and Robert A. Kosiński. "Ising-based model of opinion formation in a complex network of interpersonal interactions". In: *Physica A: Statistical Mechanics and its Applications* 361.2 (2006), pp. 651–664.

[6] Jonathan D. Power, Alexander L. Cohen, Steven M. Nelson, Gagan S. Wig, Kelly Anne Barnes, Jessica A. Church, et al. "Functional network organization of the human brain". In: *Neuron* 72.4 (2011), pp. 665–678.

[7] William Bialek, Andrea Cavagna, Irene Giardina, Thierry Mora, Edmondo Silvestri, Massimiliano Viale, et al. "Statistical mechanics for natural flocks of birds". In: *Proceedings of the National Academy of Sciences* 109.13 (2012), pp. 4786–4791.

[8] Lucas Lacasa, Bartolo Luque, Fernando Ballesteros, Jordi Luque, and Juan Carlos Nuno. "From time series to complex networks: The visibility graph". In: *Proceedings of the National Academy of Sciences* 105.13 (2008), pp. 4972–4975.

[9] Ernesto Estrada. *The structure of complex networks: theory and applications.* New York: Oxford University Press, 2012.

[10] Stefano Boccaletti, Alexander Pisarchik, Charo del Genio, and Andreas Amann. *Synchronization - From coupled systems to complex networks.* English. United Kingdom: Cambridge University Press, Mar. 2018.

[11] Duncan J. Watts and Steven H. Strogatz. "Collective dynamics of 'small-world' networks". In: *Nature* 393.6684 (1998), pp. 440–442.

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

[13] Stefano Boccaletti, Vito Latora, Yamir Moreno, Martin Chavez, and D-U Hwang. "Complex networks: Structure and dynamics". In: *Physics reports* 424.4-5 (2006), pp. 175–308.

[14] Stefano Boccaletti, Ginestra Bianconi, Regino Criado, Charo I Del Genio, Jesús Gómez-Gardenes, Miguel Romance, et al. "The structure and dynamics of multilayer networks". In: *Physics reports* 544.1 (2014), pp. 1–122.

[15] Mikko Kivelä, Alex Arenas, Marc Barthelemy, James P Gleeson, Yamir Moreno, and Mason A Porter. "Multilayer networks". In: *Journal of complex networks* 2.3 (2014), pp. 203–271.

[16] Federico Battiston, Giulia Cencetti, Iacopo Iacopini, Vito Latora, Maxime Lucas, Alice Patania, et al. "Networks beyond pairwise interactions: Structure and dynamics". In: *Physics Reports* 874 (2020), pp. 1–92.

[17] Stefano Boccaletti, Pietro De Lellis, Charo del Genio, Karin Alfaro-Bittner, Regino Criado, Sarika Jalan, et al. "The structure and dynamics of networks with higher order interactions". In: *Physics Reports* 1018 (2023).

[18] Leo Katz. "A new status index derived from sociometric analysis". In: *Psychometrika* 18.1 (1953), pp. 39–43.

[19] Linton C Freeman. "Centrality in social networks: Conceptual clarification". In: *Social networks* 1 (1978), pp. 215–239.

[20] Phillip Bonacich. "Power and centrality: A family of measures". In: *American journal of sociology* 92.5 (1987), pp. 1170–1182.

[21] Larry Page, Sergei Brin, Rajeev Motwani, and Terry Winograd. "The PageRank citation ranking: Bringing order to the Web". In: *Proceedings of the 7th International World Wide Web Conference*. 1998, pp. 161–172.

[22] David F. Gleich. "PageRank Beyond the Web". In: *SIAM Review* 57.3 (2015), pp. 321–363.

[23] Francis Bloch, Matthew O Jackson, and Pietro Tebaldi. "Centrality measures in networks". In: *Social Choice and Welfare* 61.2 (2023), pp. 413–453.

[24] Vincenzo Nicosia, Regino Criado, Miguel Romance, Giovanni Russo, and Vito Latora. "Controlling centrality in complex networks". In: *Scientific Reports* 2, 218 (Jan. 2012), p. 218.

[25] Sebastiano Vigna. "Spectral ranking". In: *Network Science* 4.4 (2016), pp. 433–445.

[26] Austin R. Benson. "Three Hypergraph Eigenvector Centralities". In: *SIAM Journal on Mathematics of Data Science* 1.2 (2019), pp. 293–312.

[27] Béla Bollobás. *Modern graph theory*. Vol. 184. Springer Science & Business Media, 2013.

[28] Vito Latora, Vincenzo Nicosia, and Giovanni Russo. *Complex Networks: Principles, Methods and Applications*. USA: Cambridge University Press, 2017.

[29] Reinhard Diestel. *Graph Theory (Graduate Texts in Mathematics)*. Springer, 2005.

[30] Jonathan L. Gross and Thomas W. Tucker. *Topological graph theory*. Courier Corporation, 2001.

[31] Norman Biggs. *Algebraic graph theory*. 67. Cambridge university press, 1993.

[32] Mark E. J. Newman. *Networks: an introduction*. Oxford; New York: Oxford University Press, 2010.

[33] Carl D. Meyer. *Matrix Analysis and Applied Linear Algebra*. SIAM, 2001.

[34] Frank Harary and Robert Z. Norman. "Some properties of line digraphs". In: *Rendiconti del Circolo Matematico di Palermo* 9.2 (May 1960), 161–168.

[35] Hassler Whitney. "Congruent Graphs and the Connectivity of Graphs". In: *American Journal of Mathematics* 54.1 (1932), pp. 150–168.

[36] Regino Criado, Julio Flores, Alejandro J. García del Amo, and Miguel Romance. "Structural properties of the line-graphs associated to directed networks". In: *Networks and Heterogeneous Media* 7.3 (2012), pp. 373–384.

[37] Gerald Alexanderson. "About the Cover: Euler and the Königsberg Bridges: A Historical View". In: *Bulletin (New Series) of the American Mathematical Society* 43 (Oct. 2006), pp. 567–573.

[38] Dorwin Cartwright and Frank Harary. "Structural balance: a generalization of Heider's theory." In: *Psychological review* 63.5 (1956), p. 277.

[39] Santo Fortunato. "Community detection in graphs". In: *Physics reports* 486.3-5 (2010), pp. 75–174.

[40] Gonzalo Contreras-Aso, Regino Criado, Guillermo Vera de Salas, and Jinling Yang. "Detecting communities in higher-order networks by using their derivative graphs". In: *Chaos, Solitons & Fractals* 177 (2023), p. 114200.

[41] Pitambar Khanra, Subrata Ghosh, David Aleja, Karin Alfaro-Bittner, Gonzalo Contreras-Aso, Regino Criado, et al. "Endowing networks with desired symmetries and modular behavior". In: *Physical Review E* 108.5 (2023).

[42] Atiyeh Bayani, Fahimeh Nazarimehr, Sajad Jafari, Kirill Kovalenko, Gonzalo Contreras-Aso, Karin Alfaro-Bittner, et al. "The transition to synchronization of networked systems". In: *Nature Communications* 15.1 (2024), p. 4955.

[43] David Aleja, Gonzalo Contreras-Aso, Karin Alfaro-Bittner, Eva Primo, Regino Criado, Miguel Romance, et al. "A compartmental model for cyber-epidemics". In: *Chaos, Solitons & Fractals* 161 (2022), p. 112310.

[44] Ming Li, Run-Ran Liu, Linyuan Lü, Mao-Bin Hu, Shuqi Xu, and Yi-Cheng Zhang. "Percolation on complex networks: Theory and application". In: *Physics Reports* 907 (2021), pp. 1–68.

[45] Dandan Li, Jing Ma, Dun Han, Mei Sun, Lixin Tian, and H Eugene Stanley. "The co-evolution of networks and prisoner's dilemma game by considering sensitivity and visibility". In: *Scientific reports* 7.1 (2017), p. 45237.

[46] *CASOS network datasets*.

[47] Jure Leskovec and Andrej Krevl. *SNAP Datasets: Stanford Large Network Dataset Collection*. http://snap.stanford.edu/data. June 2014.

[48] Jérôme Kunegis. "KONECT – The Koblenz Network Collection". In: *Proc. Int. Conf. on World Wide Web Companion*. 2013, pp. 1343–1350.

[49] Paul Erdős and Alfred Rényi. "On Random Graphs I". In: *Publicationes Mathematicae Debrecen* 6 (1959), p. 290.

[50] Albert-László Barabási. "Scale-free networks: a decade and beyond". In: *science* 325.5939 (2009), pp. 412–413.

[51] Réka Albert, Hawoong Jeong, and Albert-László Barabási. "Error and attack tolerance of complex networks". In: *nature* 406.6794 (2000), pp. 378–382.

[52] Anna Broido and Aaron Clauset. "Scale-free networks are rare". In: *Nature Communications* 10 (2019).

[53] Federico Battiston, Enrico Amico, Alain Barrat, Ginestra Bianconi, Guilherme Ferraz de Arruda, Benedetta Franceschiello, et al. "The physics of higher-order interactions in complex systems". In: *Nature Physics* 17.10 (2021), pp. 1093–1098.

[54] Mikio Nakahara. *Geometry, topology and physics*. CRC press, 2018.

[55] Ana P. Millán, Joaquín J. Torres, and Ginestra Bianconi. "Explosive higher-order Kuramoto dynamics on simplicial complexes". In: *Physical Review Letters* 124.21 (2020), p. 218301.

[56] Adilson E. Motter, Changsong Zhou, and Jürgen Kurths. "Enhancing complex-network synchronization". In: *Europhysics Letters* 69.3 (2005), p. 334.

[57] Mahdi Jalili, Ali Salehzadeh-Yazdi, Shailendra Gupta, Olaf Wolkenhauer, and Kamran Alimoghaddam. "Evolution of centrality measurements for the detection of essential proteins in biological networks". In: *Frontiers in physiology* 7 (2016), p. 166204.

[58] Jack McKay Fletcher and Thomas Wennekers. "From structure to activity: Using centrality measures to predict neuronal activity". In: *International journal of neural systems* 28.02 (2018), p. 1750013.

[59] Ivan Samoylenko, David Aleja, Eva Primo, Karin Alfaro-Bittner, Ekaterina Vasilyeva, Kirill Kovalenko, et al. "Why are there six degrees of separation in a social network?" In: *Physical Review X* 13.2 (2023), p. 021032.

[60] Ulrik Brandes. "A faster algorithm for betweenness centrality". In: *Journal of mathematical sociology* 25.2 (2001), pp. 163–177.

[61] Oskar Perron. "Zur theorie der matrices". In: *Mathematische Annalen* 64.2 (1907), pp. 248–263.

[62] Georg Ferdinand Frobenius. *Über Matrizen aus nicht negativen Elementen*. Königliche Akademie der Wissenschaften, 1912.

[63] Phillip Bonacich. "Technique for analyzing overlapping memberships". In: *Sociological methodology* 4 (1972), pp. 176–185.

[64] Crispin W. Gardiner. *Handbook of stochastic methods*. Vol. 3. Springer Berlin, 1985.

[65] Paolo Boldi, Massimo Santini, and Sebastiano Vigna. "PageRank as a Function of the Damping Factor". In: *Proceedings of the 14th International Conference on World Wide Web*. New York, NY, USA: Association for Computing Machinery, 2005, 557–566.

[66] Paolo Boldi, Massimo Santini, and Sebastiano Vigna. "PageRank: Functional dependencies". In: *ACM Trans. Inf. Syst.* 27 (Jan. 2009).

[67] Gonzalo Contreras-Aso, Regino Criado, and Miguel Romance. "Can the PageRank centrality be manipulated to obtain any desired ranking?" In: *Chaos: An Interdisciplinary Journal of Nonlinear Science* 33.8 (Aug. 2023), p. 083152.

[68] Francisco Pedroche, Miguel Romance, and Regino Criado. "A biplex approach to PageRank centrality: From classic to multiplex networks". In: *Chaos: An Interdisciplinary Journal of Nonlinear Science* 26.6 (2016), p. 065301.

[69] Konstantin Avrachenkov, Remco van der Hofstad, and Marina Sokol. "Personalized PageRank with Node-Dependent Restart". In: *Algorithms and Models for the Web Graph*. Ed. by Anthony Bonato, Fan Chung Graham, and Paweł Prałat. Cham: Springer International Publishing, 2014, pp. 23–33.

[70] Travis Martin, Xiao Zhang, and M. E. J. Newman. "Localization and centrality in networks". In: *Phys. Rev. E* 90 (5 2014), p. 052808.

[71] Florent Krzakala, Cristopher Moore, Elchanan Mossel, Joe Neeman, Allan Sly, Lenka Zdeborová, et al. "Spectral redemption: clustering sparse networks". en. In: *Proceedings of the National Academy of Sciences* 110.52 (Dec. 2013), 20935–20940.

[72] Regino Criado, Julio Flores, Esther García, Alejandro J. García del Amo, Ángel Pérez, and Miguel Romance. "On the alpha-nonbacktracking centrality for complex networks: Existence and limit cases". In: *Journal of Computational and Applied Mathematics* 350 (2019), pp. 35–45.

[73] David Aleja, Regino Criado, Alejandro J. García del Amo, Ángel Pérez, and Miguel Romance. "Non-backtracking PageRank: From the classic model to hashimoto matrices". In: *Chaos, Solitons & Fractals* 126 (2019), pp. 283–291.

[74] Luis Solá, Miguel Romance, Regino Criado, Julio Flores, Alejandro García del Amo, and Stefano Boccaletti. "Eigenvector centrality of nodes in multiplex networks". In: *Chaos: An Interdisciplinary Journal of Nonlinear Science* 23.3 (2013), p. 033131.

[75] Chinubhai G. Khatri and Calyampudi R. Rao. "Solutions to some functional equations and their applications to characterization of probability distributions". In: *Sankhyā: The Indian Journal of Statistics, Series A* (1968), pp. 167–180.

[76] Yang-Yu Liu, Jean-Jacques Slotine, and Albert-László Barabási. "Controllability of complex networks". In: *Nature* 473.7346 (2011), pp. 167–173.

[77] Francesca Arrigo, Desmond J. Higham, and Vanni Noferini. "Beyond non-backtracking: non-cycling network centrality measures". In: *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* 476.2235 (2020), p. 20190653.

[78] Esther García, Francisco Pedroche, and Miguel Romance. "On the localization of the personalized PageRank of complex networks". In: *Linear Algebra and its Applications* 439.3 (2013). Special Issue in Honor of Harm Bart, pp. 640–652.

[79] Esther García and Miguel Romance. "Controlling centrality: The inverse ranking problem for spectral centralities of complex networks". In: *Mathematical Methods in the Applied Sciences* 45.17 (2022), pp. 11755–11768.

[80] Richard M. Karp. "Reducibility among Combinatorial Problems". In: *Complexity of Computer Computations*. Ed. by Raymond E. Miller, James W. Thatcher, and Jean D. Bohlinger. Boston, MA: Springer US, 1972, pp. 85–103.

[81] Evgeniĭ Evgen'evich Tyrtyshnikov. *A brief introduction to numerical analysis*. Springer Science & Business Media, 1997.

[82] Peter D. Lax. *Linear Algebra and Its Applications*. Second. Hoboken, NJ: Wiley-Interscience, 2007.

[83] Judit Bar-Ilan. "Google bombing from a time perspective". In: *Journal of Computer-Mediated Communication* 12.3 (2007), pp. 910–938.

[84] Konstantin Avrachenkov and Nelly Litvak. "The Effect of New Links on Google Pagerank". In: *Stochastic Models* 22.2 (2006), pp. 319–331.

[85] Martin Olsen. "Maximizing PageRank with New Backlinks". In: *Algorithms and Complexity*. Ed. by Tiziana Calamoneri and Josep Diaz. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 37–48.

[86] Daniel Sheldon. "Manipulation of PageRank and Collective Hidden Markov Models". PhD thesis. Cornell University, 2010.

[87] Olivier Fercoq, Marianne Akian, Mustapha Bouhtou, and Stephane Gaubert. "Ergodic Control and Polyhedral Approaches to PageRank Optimization". In: *IEEE Transactions on Automatic Control* 58.1 (2013), pp. 134–148.

[88] Balázs Csanád Csáji, Raphaël M Jungers, and Vincent D Blondel. "PageRank optimization by edge selection". In: *Discrete Applied Mathematics* 169 (2014), pp. 73–87.

[89] David Aleja, Julio Flores, Eva Primo, and Miguel Romance. "Time-dependent personalized PageRank for temporal networks: Discrete and continuous scales". In: *Chaos: An Interdisciplinary Journal of Nonlinear Science* 34.8 (Aug. 2024), p. 083145.

[90] Julie Morrison, Rainer Breitling, Desmond Higham, and David Gilbert. "GeneRank: Using search engine technology for the analysis of microarray experiments". In: *BMC bioinformatics* 6 (Feb. 2005), p. 233.

[91] Valerio Freschi. "Protein function prediction from interaction networks using a random walk ranking algorithm". In: *2007 IEEE 7th International Symposium on BioInformatics and BioEngineering*. 2007, pp. 42–48.

[92] Jia-Yu Pan, Hyung-Jeong Yang, Christos Faloutsos, and Pinar Duygulu. "Automatic Multimedia Cross-Modal Correlation Discovery". In: *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. KDD '04. Seattle, WA, USA: Association for Computing Machinery, 2004, 653–658.

[93] Francisco Pedroche, Esther García, Miguel Romance, and Regino Criado. "Sharp estimates for the personalized Multiplex PageRank". In: *Journal of Computational and Applied Mathematics* 330 (2018), pp. 1030–1040.

[94] Roger A. Horn and Charles R. Johnson. *Topics in Matrix Analysis*. Cambridge University Press, 1991.

[95] Sinan G. Aksoy, Ilya Amburg, and Stephen J. Young. *Scalable tensor methods for nonuniform hypergraphs*. 2024.

[96] Kung Ching Chang, Kelly J. Pearson, and Tan Zhang. "Perron-Frobenius theorem for nonnegative tensors". In: *Communications in Mathematical Sciences* 6.2 (2008), pp. 507 –520.

[97] Liqun Qi and Ziyan Luo. *Tensor Analysis: Spectral Theory and Special Tensors*. Philadelphia, PA: Society for Industrial and Applied Mathematics, 2017.

[98] Kung Ching Chang, Kelly J. Pearson, and Tan Zhang. "Some variational principles for Z-eigenvalues of nonnegative tensors". In: *Linear Algebra and its Applications* 438.11 (2013), pp. 4166–4182.

[99] Michael Ng, Liqun Qi, and Guanglu Zhou. "Finding the largest eigenvalue of a nonnegative tensor". In: *SIAM Journal on Matrix Analysis and Applications* 31.3 (2010), pp. 1090–1099.

[100] Kirill Kovalenko, Miguel Romance, Ekaterina Vasilyeva, David Aleja, Regino Criado, Daniil Musatov, et al. "Vector centrality in hypergraphs". In: *Chaos, Solitons & Fractals* 162 (2022), p. 112397.

[101] Jean-Claude Bermond, Marie-Claude Heydemann, and Dominique Sotteau. "Line graphs of hypergraphs I". In: *Discrete Mathematics* 18.3 (1977), pp. 235–241.

[102] Gonzalo Contreras-Aso, Cristian Pérez-Corral, and Miguel Romance. "Uplifting edges in higher order networks: spectral centralities for non-uniform hypergraphs". In: *arXiv:2310.20335* (2023).

[103] Yaoming Zhen and Junhui Wang. "Community Detection in General Hypergraph Via Graph Embedding". In: *Journal of the American Statistical Association* 0.0 (2022), pp. 1–10.

[104] Xavier Ouvrard, Jean-Marie Le Goff, and Stéphane Marchand-Maillet. *Adjacency and Tensor Representation in General Hypergraphs Part 1: e-adjacency Tensor Uniformisation Using Homogeneous Polynomials*. 2018.

[105] Michael Ng, Liqun Qi, and Guanglu Zhou. "Finding the Largest Eigenvalue of a Nonnegative Tensor". In: *SIAM J. Matrix Analysis Applications* 31 (Jan. 2009), pp. 1090–1099.

[106] Anirban Banerjee, Arnab Char, and Bibhash Mondal. "Spectra of general hypergraphs". In: *Linear Algebra and its Applications* 518 (2017), pp. 14–30.

[107] Nicholas W. Landry, Maxime Lucas, Iacopo Iacopini, Giovanni Petri, Alice Schwarze, Alice Patania, et al. "XGI: A Python package for higher-order interaction networks". In: *Journal of Open Source Software* 8.85 (May 2023), p. 5162.

[108] Austin R. Benson, Rediet Abebe, Michael T. Schaub, Ali Jadbabaie, and Jon Kleinberg. "Simplicial closure and higher-order link prediction". In: *Proceedings of the National Academy of Sciences* (2018).

[109] Lorenzo Isella, Juliette Stehlé, Alain Barrat, Ciro Cattuto, Jean-François Pinton, and Wouter Van den Broeck. "What's in a crowd? Analysis of face-to-face behavioral networks". In: *Journal of theoretical biology* 271.1 (2011), pp. 166–180.

[110] Douglas R. Hofstadter. *Gödel, Escher, Bach: an Eternal Golden Braid*. Basic Books Inc., 1979.

[111] Juliette Stehlé, Nicolas Voirin, Alain Barrat, Ciro Cattuto, Lorenzo Isella, Jean-François Pinton, et al. "High-resolution measurements of face-to-face contact patterns in a primary school". In: *PloS one* 6.8 (2011), e23176.

[112] Rossana Mastrandrea, Julie Fournet, and Alain Barrat. "Contact patterns in a high school: a comparison between data collected using wearable sensors, contact diaries and friendship surveys". In: *PloS one* 10.9 (2015), e0136497.

[113] Ciro Cattuto, Wouter Van den Broeck, Alain Barrat, Vittoria Colizza, Jean-François Pinton, and Alessandro Vespignani. "Dynamics of person-to-person interactions from distributed RFID sensor networks". In: *PloS one* 5.7 (2010), e11596.

[114] Kwang-Il Goh, Michael E Cusick, David Valle, Barton Childs, Marc Vidal, and Albert-László Barabási. "The human disease network". In: *Proceedings of the National Academy of Sciences* 104.21 (2007), pp. 8685–8690.

[115] Giorgio Gallo, Giustino Longo, Stefano Pallottino, and Sang Nguyen. "Directed hypergraphs and applications". In: *Discrete Applied Mathematics* 42.2 (1993), pp. 177–201.

[116] Steffen Klamt, Utz-Uwe Haus, and Fabian J. Theis. "Hypergraphs and Cellular Networks." In: *PLoS Comput. Biol.* 5.5 (2009).

[117] V. Wakelam, E. Herbst, J.-C. Loison, I. W. M. Smith, V. Chandrasekaran, B. Pavone, et al. "A KInetic Database for Astrochemistry (KIDA)". In: *The Astrophysical Journal Supplement Series* 199.1 (Mar. 2012), p. 21.

[118] Zachary A. King, Justin Lu, Andreas Dräger, Philip Miller, Stephen Federowicz, Joshua A. Lerman, et al. "BiGG Models: A platform for integrating, standardizing and sharing genome-scale models". In: *Nucleic Acids Research* 44.D1 (Oct. 2015), D515–D522.

[119] Regino Criado, Miguel Romance, and María Vela-Pérez. "Hyperstructures, a new approach to complex systems". In: *International Journal of Bifurcation and Chaos* 20.03 (2010), pp. 877–883.

[120] Qing Xu, Lizhu Sun, and Changjiang Bu. "The two-steps eigenvector centrality in complex networks". In: *Chaos, Solitons & Fractals* 173 (2023), p. 113753.

[121] Luca Gallo, Riccardo Muolo, Lucia Valentina Gambuzza, Vito Latora, Mattia Frasca, and Timoteo Carletti. "Synchronization induced by directed higher-order interactions". In: *Communications Physics* 5.1 (2022), p. 263.

[122] Jinshan Xie and Liqun Qi. "Spectral directed hypergraph theory via tensors". In: *Linear and Multilinear Algebra* (Jan. 2016), pp. 1–15.

[123] Pieter P. Plehiers, Guy B. Marin, Christian V. Stevens, and Kevin M. Van Geem. "Automated reaction database and reaction network analysis: extraction of reaction templates using cheminformatics". In: *Journal of Cheminformatics* 10.1 (Mar. 2018).

[124] Nicholas W. Landry, Maxime Lucas, Iacopo Iacopini, Giovanni Petri, Alice Schwarze, Alice Patania, et al. "XGI: A Python package for higher-order interaction networks". In: *Journal of Open Source Software* 8.85 (May 2023), p. 5162.

[125] Gonzalo Contreras-Aso, Regino Criado, and Miguel Romance. "Beyond directed hypergraphs: heterogeneous hypergraphs and spectral centralities". In: *Journal of Complex Networks* 12.4 (Sept. 2024), cnae037.

[126] Charles H. Hubbell. "An input-output approach to clique identification". In: *Sociometry* (1965), pp. 377–399.

[127] Jon M. Kleinberg. "Authoritative Sources in a Hyperlinked Environment". In: *J. ACM* 46.5 (1999), 604–632.

[128] Abraham Berman and Robert J. Plemmons. *Nonnegative Matrices in the Mathematical Sciences*. Society for Industrial and Applied Mathematics, 1994.

[129] Garrett Birkhoff and Richard S. Varga. "Reactor criticality and nonnegative matrices". In: *Journal of the Society for Industrial and Applied Mathematics* 6.4 (1958), pp. 354–377.

[130] Pablo Tarazaga, Marcos Raydan, and Ana Hurman. "Perron–Frobenius theorem for matrices with some negative entries". In: *Linear Algebra and its Applications* 328.1-3 (2001), pp. 57–68.

[131] Kelly J. Pearson and Tan Zhang. "On spectral hypergraph theory of the adjacency tensor". In: *Graphs and Combinatorics* 30 (2014), pp. 1233–1248.

[132] Tamara G. Kolda and Jackson R. Mayo. "An Adaptive Shifted Power Method for Computing Generalized Tensor Eigenpairs". In: *SIAM Journal on Matrix Analysis and Applications* 35.4 (2014), pp. 1563–1581.

[133] Austin Benson and David Gleich. "Computing Tensor Z-Eigenvectors with Dynamical Systems". In: *SIAM Journal on Matrix Analysis and Applications* 40 (Nov. 2019), pp. 1311–1324.

[134] Jerri L. Ledford. *Search engine optimization bible*. Vol. 584. John Wiley & Sons, 2015.

Summary in Spanish/Resumen en castellano

Esta tesis culmina el trabajo doctoral llevado a cabo en el área de Matemática Aplicada de la Universidad Rey Juan Carlos durante tres años. En la presente Sección resumimos en castellano los antecedentes, objetivos, metodología, resultados y conclusiones de este trabajo, en cumplimiento con la normativa vigente de la Escuela Internacional de Doctorado de la Universidad Rey Juan Carlos.

Antecedentes

La ciencia de las redes complejas ha experimentado un auge vertiginoso en las últimas dos décadas, cautivando a investigadores de diversos campos por su capacidad para modelar y comprender una amplia gama de fenómenos del mundo real [13, 9]. Si bien sus orígenes se remontan a trabajos pioneros en física y matemáticas en el siglo XX, es en las últimas décadas que este campo ha experimentado un crecimiento exponencial, impulsado por avances tecnológicos y la necesidad de abordar problemas cada vez más complejos en diversas áreas del conocimiento.

Uno de los principales campos de estudio dentro de este área es el estudio de medidas de centralidad en redes [23]: identificar en base a algún criterio de importancia cuáles son los nodos más relevantes en una red, cuantitativamente. El origen de este área de hecho trasciende al de las redes complejas, siendo inicialmente desarrollado por sociólogos interesados en cuantificar interacciones entre individuos [18]. Las medidas de centralidad ganaron una gran cantidad de atención con el éxito de PageRank [21], la medida de centralidad usada por Google para obtener un ranking the las páginas de Internet más relevantes.

Gran cantidad de tinta se ha vertido en el estudio de medidas de centralidad, principalmente definiendo medidas en base a nuevos criterios de importancia, pero también analizando y clasificando las ya existentes. Dentro de esta clasificación, un tipo especialmente relevante es el de las medidas de centralidades espectrales [25]: estas están fundamentadas en el álgebra lineal, y en concreto hacen referencia a propiedades espectrales de matrices que caracterizan la red. De especial relevancia es el Teorema de Perron-Frobenius [61, 62], que garantiza existencia y unicidad de un autovector positivo en matrices no negativas irreducibles, que está en la base de muchas de estas medidas (por ejemplo, de PageRank). Sin embargo algo que siempre quedó de lado, salvo por unos pocos casos excepcionales [24, 79], es el estudio de las posibilidades de control de estas medidas, es decir, qué cambios hemos de realizar en la red o los parámetros de la medida para obtener cualquier centralidad a nuestro antojo. Lo más parecido en esta dirección se correspondería con el *Search Engine Optimization* (SEO), pero son de aplicabilidad limitada, especialmente fuera del ámbito del posicionamiento de páginas web [134].

Volviendo a las redes complejas, inicialmente, el foco de atención se centró en redes estándar, estructuras relativamente simples compuestas por nodos y enlaces que las conectan. Estas redes, inspiradas en modelos matemáticos como la teoría de grafos, se utilizaron para estudiar una amplia gama de fenómenos, desde la estructura de las redes sociales hasta la propagación de enfermedades infecciosas, o la identificación de

elementos importantes mediante las medidas de centralidad previamente mencionadas. Sin embargo, a medida que la complejidad de los sistemas que se buscaban modelar aumentaba, surgió la necesidad de abstracciones matemáticas más sofisticadas.

En este contexto, las redes multicapa [14, 15] y los hipergrafos [16, 17] se erigieron como herramientas invaluables para capturar la intrincada organización de sistemas interconectados. Las redes multicapa introducen una nueva dimensión al permitir la existencia de múltiples tipos de relaciones entre los nodos, reflejando así la riqueza y diversidad de las interacciones que observamos en el mundo real. Por otro lado, los hipergrafos van un paso más allá, permitiendo que un solo enlace conecte a más de dos nodos, lo que los convierte en una herramienta ideal para modelar sistemas con relaciones de alta complejidad.

Esta capacidad de generalizar es un arma de doble filo, puesto que viene a costa de, también, una mayor complejidad en las herramientas disponibles. En el caso de los hipergrafos y sus propiedades espectrales, esto pasa por abandonar el ámbito del álgebra lineal por el del álgebra multilineal [97], o incluso el de la topología algebraica [54] en ciertos tipos de hipergrafos. Es por esto que las principales generalizaciones de la centralidad de autovector, la centralidad espectral por excelencia en redes, propuestas en [26], tienen grandes limitaciones: solo están definidas para hipergrafos uniformes, y no dirigidos.

Objetivos

Los objetivos principales de esta tesis son:

- Proporcionar una visión general exhaustiva sobre las posibilidades de controlar diversas medidas de centralidad del tipo espectral. Como parte de ello, se establecerá una clasificación de los paradigmas de control de medidas de centralidad en función de dos ejes: "tipo" de control (estructural o paramétrico) y "cantidad" de control (completo, ranking o localización).

  Se comprobará cuánto control otorga cada tipo de modificación sobre la centralidad de la red, en función del tipo de red y la medida de centralidad considerados.

- Extender y generalizar las centralidades espectrales en hipergrafos, en concreto la propuesta de Benson [26]. Esto se debe a que los métodos existentes presentan limitaciones, como la incapacidad de aplicarlos a hipergrafos no uniformes y dirigidos/heterogéneos. Se busca abordar estas limitaciones y proponer alternativas analíticamente consistentes y numéricamente sólidas basadas en la teoría de Perron-Frobenius para tensores.

- Evaluar la capacidad de control de estas nuevas propuestas de centralidad espectral en hipergrafos, replicando parte de los análisis desarrollados para redes estándar.

Metodología

El enfoque de esta tesis es principalmente teórico, centrado en la comprensión más que en la aplicación. Para lograr los objetivos planteados, se llevará a cabo la siguiente metodología:

1. Revisión y análisis de medidas de centralidad espectral: Se revisarán y analizarán diversas medidas de centralidad espectral propuestas para grafos, redes multicapa y hipergrafos en la literatura. Esto permitirá comprender sus características, funcionamiento y limitaciones.

2. Exploración de escenarios de control: Se explorarán diferentes escenarios de control para las medidas de centralidad espectral, como el ajuste de pesos y la adición de bucles. Se analizará cómo estas técnicas pueden utilizarse para alterar las medidas y obtener resultados específicos, utilizando técnicas de álgebra lineal para obtener conclusiones objetivas y cuantitativas.

3. Desarrollo de extensiones y generalizaciones de centralidades espectrales en hipergrafos: Se desarrollarán extensiones y generalizaciones de las centralidades espectrales para hipergrafos no uniformes y dirigidos/heterogéneos. Esto se logrará aprovechando la teoría de Perron-Frobenius para tensores, permitiendo superar las limitaciones de los métodos existentes.

4. Evaluación de la capacidad de control de las nuevas propuestas: Se evaluará la capacidad de control de las nuevas propuestas de centralidad espectral en hipergrafos mediante el ajuste de pesos. Esto permitirá determinar su efectividad práctica en la manipulación de las medidas.

Resultados

Se han obtenido los siguientes resultados:

1. Se ha presentado un análisis exhaustivo de las posibilidades de control de medidas de centralidad espectral en redes estándar y multiplex. Para la centralidad de autovector, se han estudiado los efectos de diferentes técnicas como el ajuste de pesos, la adición de aristas o la adición de bucles. Esta última ha proporcionado un índice de controlabilidad, calculado explícitamente en redes reales. Se ha estudiado la relación entre las centralidades de la red y estructuras como el *line graph*, comprobando su falta de control.

   Para la centralidad PageRank se han estudiado similares cuestiones, pero además la presencia de parámetros en esta medida ha permitido estudiar cómo afecta su modificación a la centralidad resultante, entendiendo tanto las posibilidades de control total, como de control de ranking y localización. Lo mismo se ha podido estudiar en otras variantes de dicha centralidad, como en la node-dependent restart PageRank.

2. Se han propuesto extensiones y generalizaciones analíticamente consistentes y numéricamente sólidas de las centralidades espectrales para hipergrafos no uniformes. Concretamente, en el caso no dirigido se ha propuesto el uso de una operación, el *uplift*, para la uniformización de un hipergrafo general, de forma que las técnicas estándar de centralidad se pueden usar de forma consistente.

3. Se ha propuesto una taxonomía de hipergrafos, distinguiendo entre tipos en base a las simetrías de sus tensores de adjacencia. En ese sentido, se llaman hipergrafos heterogéneos a todos aquellos que no sean no dirigidos. Dentro de los hipergrafos heterogéneos encontramos algunos tipos especialmente relevantes, como los tradicionalmente considerados como dirigidos, pero también otros nuevos como los cíclicos.

4. Para varios tipos relevantes de hipergrafos heterogéneos se han generalizado las medidas de centralidad espectral, manteniendo siempre la consistencia con la teoría de Perron-Frobenius para tensores, y se ha comprobado su efectividad mediante comparaciones numéricas en hipergrafos construidos con reales.

5. Se ha evaluado la capacidad de control de las nuevas propuestas de centralidad espectral en hipergrafos heterogéneos mediante el ajuste de pesos, comprobando la posibilidad de obtener control total en el caso dirigido, pero sin posibilidad de control en el resto de casos.

Conclusiones

Este trabajo teórico sienta las bases para el desarrollo y evaluación de nuevas medidas de centralidad. Los métodos desarrollados pueden servir como campo de pruebas para evaluar la capacidad de control de nuevas medidas y las técnicas empleadas tienen un alcance más general, ya que abordan las propiedades espectrales de objetos matemáticos, no solo redes específicas.

Aportaciones. Las principales aportaciones son:

- Se presenta un análisis exhaustivo de las posibilidades de controlar diversas medidas de centralidad espectral, incluyendo los efectos de diferentes técnicas como el ajuste de pesos, la adición de bucles y la modificación de parámetros.

- Se proponen extensiones y generalizaciones analíticamente consistentes y numéricamente sólidas de las centralidades espectrales para hipergrafos no uniformes y dirigidos/heterogéneos. Estas propuestas se basan en la teoría de Perron-Frobenius para tensores y abordan las limitaciones de los métodos existentes.

- Se evalúa la capacidad de control de las nuevas propuestas de centralidad espectral en hipergrafos mediante el ajuste de pesos. Se presentan resultados numéricos que demuestren la efectividad de estas propuestas en la manipulación de las medidas.

Implicaciones. Las principales implicaciones son:

- Las investigaciones presentadas en esta tesis pueden servir como base para el desarrollo de nuevas medidas de centralidad que sean más sensibles a las características específicas de las redes y que tengan una mayor capacidad de control.
- Los métodos desarrollados para controlar las medidas de centralidad espectral pueden ser aplicados a todo tipos de redes, como redes sociales, redes biológicas y redes de infraestructuras, ya que son agnósticos al tipo concreto por el caracter matemático de los mismos.
- La comprensión de las propiedades espectrales de las redes puede ayudar a identificar nodos y subredes importantes en la red, lo que puede tener aplicaciones en diversas áreas, como la optimización de redes, la detección de anomalías y la difusión de información.

Futuro. Se abre un abanico de posibilidades para futuras investigaciones que ahondan en los resultados obtenidos, o que utilizan las técnicas presentadas en nuevos escenarios.

- Extender los resultados a otros tipos de medidas de centralidad espectral que no se han considerado en esta tesis, como podrían ser la centralidad de Katz o la α-centralidad. Así mismo, gran parte de las investigaciones en cuanto a control estructural se han referido a modificaciones de los pesos, dado su alto grado de manejo, sin embargo está por ver cómo afectan muchos de los otros paradigmas de control estructural introducidos al resto de medidas.
- Estudiar la aplicabilidad de los métodos desarrollados a redes reales en diferentes dominios, como las redes sociales, las redes biológicas y las redes de infraestructuras. En este sentido, hasta ahora nos hemos preocupado de realizar análisis teóricos en abstracto, pero es interesante deducir las consecuencias de dichos análisis en redes reales, y sus posibles aplicaciones.
- Explorar la relación entre las centralidades espectrales y la dinámica de redes e hipergrafos, como la propagación de epidemias o la sincronización de unidades. Debería ser posible relacionar los efectos de distintas centralidades obtenidas para una red y una medida de centralidad mediante las técnicas de control aquí establecidas, y las consecuencias para las dinámicas que se establezcan sobre la red.

En definitiva, este trabajo de investigación contribuye a la comprensión de las medidas de centralidad espectral en redes complejas y sienta las bases para el desarrollo de nuevas herramientas y métodos para el análisis y control de redes.