Université catholique de Louvain
Ecole polytechnique de Louvain
Département d'ingénierie mathématique

Ranking Large Networks:

Leadership, Optimization and Distrust

Cristobald de Kerchove d'Exaerde

Thèse présentée en séance publique le 27 août 2009
en vue de l'obtention du grade de
Docteur en Sciences appliquées.

Promoteurs : Vincent Blondel et Paul Van Dooren

Jury :
Marco Saerens, président, Université catholique de Louvain
Vincent Blondel, Université catholique de Louvain
Jean-Loup Guillaume, Université Paris 6, France
Ilse Ipsen, North Carolina State University, USA
Paul Van Dooren, Université catholique de Louvain

Acknowledgments

First of all, I would like to thank my two supervisors Paul Van Dooren and Vincent Blondel who carefully guided me during the five years of my thesis. I really appreciated their attention, their availability and their support every time I needed it. I owe my different research areas and my publications to them.

This thesis has also been reviewed by three members of my thesis committee, namely Ilse Ipsen, Jean-Loup Guillaume and Marco Saerens. They largely improved the quality of the manuscript and they gave me helpful advices during my research. I thank Ilse Ipsen for her nice welcome in the North Carolina State University and for her detailed feedback on the thesis. I also want to thank Jean-Loup Guillaume for the friendly interactions we had during and after his postdoctoral research in the *Université catholique de Louvain*. I am thankful to Marco Saerens who accepted to chair my private defense and who usefully provided me with his expertise in data mining.

It was a great pleasure to collaborate with many different persons in the Department of mathematical engineering of the Université catholique de Louvain. I enjoyed the joint work with Laure Ninove about PageRank maximization that led to a chapter in this thesis. She had the talent to make everything interesting. I also thank a lot the previous and actual members of the *Large Graphs and Networks* team for their comments, remarks and ideas during the seminars and the scientific retreats. I give at the end of Chapter 1 a complete list of coauthors and I thank them for their collaboration.

CESAME is a great place where the atmosphere is warm and friendly. I would like to thank especially the secretaries who contribute to a large extent to the good ambiance of the department and who kindly help me for the administrative tasks. I will never forget Jean-Baptiste Coulaud who shared my office and who was always ready to help me for mathematical and computer issues.

This thesis was prepared at the Department of mathematical engineering of the Université catholique de Louvain, in the Center for System Engineering and Applied Mechanics. During the preparation of my PhD, I was supported financially by research programs ARC Large Graphs and Networks, an Action de recherche concertée of the Communauté Française de Belgique, and IAP DYSCO (Dynamical Systems, Control, and Optimization), a Belgian research network funded by the Interuniversity Attraction Poles Programme, initiated by the Belgian State, Science Policy Office.

The results on mobile phone networks have been possible thanks to a close collaboration with the company IKAN consulting. Moreover Laure Ninove and Maxime Melchior allowed me to respectively insert the figures 2.1 and 3.10 that appropriately illustrate the thesis.

Last but not least, I am deeply grateful to my family, particularly my wife Laurence and my children, for their support and their patience with my moments of absent-mindedness due to some proofs in my thesis.

Summary

The present work is devoted to the analysis of large networks that mathematically models the connections between items (generally millions or billions of items). For instance, people using their mobile phones *live* in a network where they are connected by their calls, SMSs, MMSs, etc. Another example is given by the set of webpages in the World Wide Web that are connected by their hyperlinks. The representation of these large databases by networks allows us to extract different type of hidden information as the sets of communities and the importance of the nodes in the network. We investigate the second issue where the target is to assign a rank to all nodes that will be relevant for classifying them by order of importance in some context. We will consider five topics related to ranking methods for networks:

1. Ranking can be used to identify the leaders among the customers of a mobile phone company, that is, people who has the capacity to influence their contacts. Surprisingly enough, the study shows that some measure based on the structural position of a customer in the mobile phone network provides relevant information to identify leaders. For example, one of the interesting measures is given by the *social leaders* who are the customers having more pairs of friends than their friends. The quality of a leader is then measured by observing the MMS activations among the customers.

2. We call a *degree leader* someone who has more friends than his friends. We could expect that when your number of friends increases, the probability to be degree leader becomes higher. That question is analyzed for random networks that have the same degree distribution than real social networks, i.e., the proportion of nodes of degree d follows a power law $\sim d^{-\gamma}$ with $2 < \gamma < 3$. We

prove that the probability does not necessarily increases with d, more precisely it increases or decreases depending on a threshold in the power γ.

3. A well-known example of ranking method is given by the Google's PageRank that lists the webpages according to their relative credibility. We consider the problem of maximizing the average PageRank of a set of webpages when we consider two realistic constraints: one can only control the hyperlinks of one's webpages and one must points to the rest of the web. We then prove that the optimal website necessarily has a particular structure with a single webpage pointing to the rest of the web.

4. The PageRank (and many other ranking methods) is based on a random walk over the network. At each step, the walker visits a new node in the network by following one of the possible connections of the previous node. If a node is often visited by the walker, then its rank will be high. We extend the random walk to the case where connections of distrust are present in the network. The idea is to update a list of forbidden nodes according to these negative connections so that the walker has a memory to avoid distrusted nodes. Several ranking methods based on that new random walk are proposed and illustrated.

5. Opinions can be modeled by a network where the connections are directed and weighted by a vote. A first rank, called reputation, is assigned to every node according to the weights of their incoming connections, that is, the votes received from other nodes. In addition, a second rank, called credibility, is also assigned to every node and depends on the outgoing connections that allows to measure the bias of the given votes. We propose an iterative filtering that alternatively uses the credibility of the votes to determine the reputations of the nodes, and then the reputation of the nodes to update the real bias of the votes. We prove that the iterations always converge to a unique solution.

Contents

List of Notation

$\mathbb{N}$	set of natural numbers
$\mathbb{N}_0$	set of nonnegative integers
$\mathbb{R}$	set of real numbers
$\mathbb{R}_{\geq 0}$	set of nonnegative real numbers
$\mathbb{R}_{>0}$	set of positive real numbers
$\emptyset$	empty set
$\mathcal{P}$	polytope in $\mathbb{R}^n$
$\mathcal{H}$	hypercube in $\mathbb{R}^n$
$\mathcal{N}$	set of nodes $\{1, \ldots, n\}$
$\mathcal{B}$	set of nodes in a blacklist, $\mathcal{B} \subseteq \mathcal{N}$
$\mathcal{L}$	set of links $\mathcal{L} \subseteq \mathcal{N} \times \mathcal{N}$
$\mathcal{G} = (\mathcal{N}, \mathcal{L})$	network defined by a set of nodes $\mathcal{N}$ and a set of links $\mathcal{L}$
(i, j)	link from node i to node j
$i \to j$	i is in the set of parents of j
$i \leftrightarrow j$	i is in the set of neighbors of j
$j \leftarrow i$	j is in the set of children of i
$\boldsymbol{d}_i$	outdegree of node i
	or in Chapter 7, belief divergence of rater i
d_{max}	maximal outdegree
d_{min}	minimal outdegree
z	average outdegree
$\mathcal{L}_{\mathcal{I}}$	set of internal links, i.e. $\{(i, j) \in \mathcal{L} \colon i, j \in \mathcal{I}\}$
$\mathcal{L}_{\text{out}(\mathcal{I})}$	set of external outlinks, i.e. $\{(i, j) \in \mathcal{L} \colon i \in \mathcal{I}, j \notin \mathcal{I}\}$
$\mathcal{L}_{\text{in}(\mathcal{I})}$	set of external inlinks, i.e. $\{(i, j) \in \mathcal{L} \colon i \notin \mathcal{I}, j \in \mathcal{I}\}$
$\mathcal{L}_{\overline{\mathcal{I}}}$	set of external links, i.e. $\{(i, j) \in \mathcal{L} \colon i, j \notin \mathcal{I}\}$
$\mathcal{G}_{\mathcal{I}}$	subnetwork of $\mathcal{G}$ determined by the set $\mathcal{I}$, i.e., $(\mathcal{I}, \mathcal{L}_{\mathcal{I}})$

A	adjacency matrix
S, P	row stochastic matrices
I	identity matrix
$B, X, \dots$	matrices
B^T	transpose of the matrix B
$B_{\mathcal{I},\mathcal{J}}$	submatrix of B induced by the index sets $\mathcal{I}, \mathcal{J}$
$B_{\mathcal{I}}$	principal submatrix of B induced by $\mathcal{I}$
$\rho(B)$	spectral radius of the matrix B
$\lVert\cdot\rVert$	any vector norm operator
$\lVert\cdot\rVert_1$	ℓ_1 vector norm, i.e. sum vector norm
$\lVert\cdot\rVert_2$	ℓ_2 vector norm, i.e. Euclidean vector norm
$\mathrm{int}(\mathcal{H})$	interior of the set $\mathcal{H}$
$\mathrm{Prob}(X\|Y)$	probability of event X given Y
$B_1 \circ B_2$	componentwise product of the matrices B_1 and B_2
$\frac{[B_1]}{[B_2]}$	componentwise division of the matrix B_1 by B_2
$\mathbf{1}$	vector of all ones
0	zeros scalar, vector or matrix
$\boldsymbol{e}_i$	i^{th} standard basis vector in $\mathbb{R}^n$
$\boldsymbol{e}_{\mathcal{I}}$	vector with a 1 in the entries of $\mathcal{I}$ and a 0 elsewhere
$\boldsymbol{z}$	personalization vector in $\mathbb{R}^n$, i.e., $\boldsymbol{z}_i > 0$ and $\lVert\boldsymbol{z}\rVert_1 = 1$
$\boldsymbol{n}_i$	number of votes of the rater i
$\boldsymbol{x}, \boldsymbol{y}, \dots$	vectors
$\boldsymbol{x}^T$	transpose of the vector $\boldsymbol{x}$
$\boldsymbol{x}_{\mathcal{I}}$	subvector of $\boldsymbol{x}$ determined by the index set $\mathcal{I}$
$\boldsymbol{x} \neq \boldsymbol{y}$	componentwise inequalities, i.e., $\boldsymbol{x}_i \neq \boldsymbol{y}_i$ for all $i \in \mathcal{N}$
$(\boldsymbol{x}^t)$	sequence of vectors, i.e. $\{\boldsymbol{x}^t : t \geq 0\}$
(X_t)	sequence of matrices, i.e. $\{X_t : t \geq 0\}$
$(\mathcal{B}_t)$	sequence of blacklists, i.e. $\{\mathcal{B}_t : t \geq 0\}$
$\nabla_{\boldsymbol{x}} E(\tilde{\boldsymbol{x}})$	gradient of the scalar function $E(\boldsymbol{x})$ evaluated in $\tilde{\boldsymbol{x}}$

List of Figures

Chapter 1

General Introduction

In the last few years, the study of networks has received an enormous amount of attention from the scientific community [8, 78], in disciplines as diverse as biology (metabolic and protein interaction), computer and information sciences (the Internet and the World Wide Web), etc. The present work fits in the context of **information retrieval** in large networks. The information one extracts may concern many different aspects of the network. For instance, the modularity of the network [80] measures how much a network is divisible into communities, its diameter [3] indicates the maximal distance between two nodes or yet, for every node, the PageRank [85] gives a rank proportional to the probability of presence of a walker moving randomly through the network. This thesis focuses on **ranking measures** like the PageRank that allows us to classify the nodes of a network by order of importance.

The extraction of the information is the **algorithmic part** of the problem. The increasing size of databases about documents, customers, emails, etc. is accompanied with the need of fast algorithms to extract useful information. That explains why we use the term *large* networks, they correspond to networks where the number of nodes – generally several millions or more – requires efficient algorithms in time and in memory space.

Finally, the information is extracted from structured data, that is the **network**. Each node of the network represents an item that is possibly connected to some other node by the links that can be undirected, directed and weighted. For example, these items will be in the next chapters the customers of a Belgian mobile phone company, the

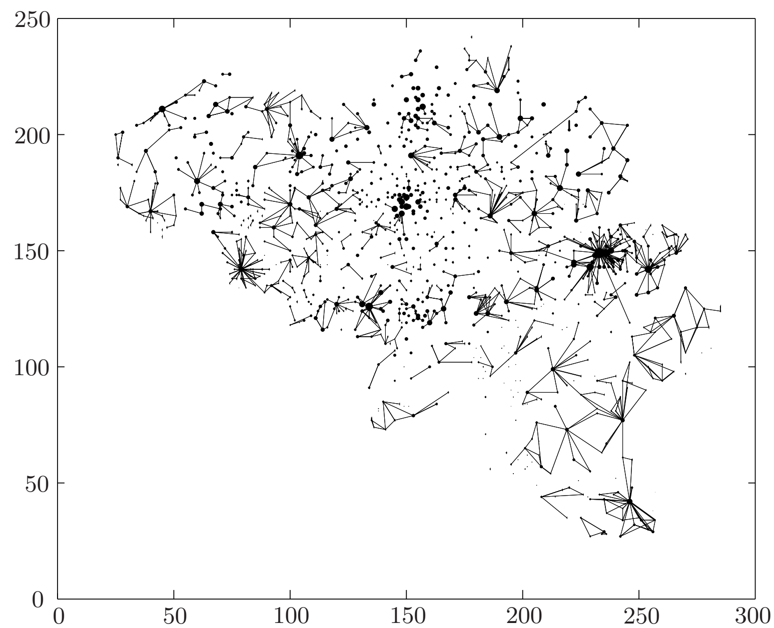


Figure 1.1: Network representation where the nodes are the municipalities, the size of the node is proportional to the number of customers in the municipality and links are drawn between two municipalities if one of them accounts for at least 5% of calls from the other municipality.

webpages crawled by Google or the raters of the website Epinion. These nodes are respectively connected by their hyperlinks, their calls and their ratings. Another example is illustrated in Fig. 1.1 and comes from the aggregation of the network of the customers of a Belgian mobile phone company around the municipalities. All these examples are *social networks*, meaning that the links in these networks are somehow related to social relationships. The network of webpages is also considered as an *information network*, meaning that the links connect sources of information. The rest of this thesis mainly deals with such types of networks and with questions about *leadership*, about *optimization* of ranking measures and about the consideration of links with *negative weights*.

The **next sections** explain the advance and the development of the ideas that led to the different chapters of this thesis. We also describe the types of information we looked for, the algorithms we applied to those and the items mathematically represented by networks. The first section gives the context originated from the different research topics and the connections between them. The content of the chapters are then detailed in the second section that gives a fortaste of the main issues. The last section surveys the publications made since the beginning of the thesis and relates them to this thesis.

1.1 Context of the thesis

The topics studied in this work are connected by several questions that arise from several contexts. We present here what leads us to consider the problems exposed in the chapters of this thesis.

The research contract. The beginning of this work was firstly motivated by a research contract with a Belgian mobile phone company for which the results are given in Chapter 3. They supplied us with a large amount of data representing a list of customers and their interactions by calls, text messages and MMSs – Multimedia Messaging Services – during a period of several months. The goal was then to detect important nodes in the context of viral marketing [13]. This refers to marketing techniques that seek to exploit pre-existing networks to produce exponential increases in brand awareness, through viral processes similar to the spread of an epidemic. It is word-of-mouth delivered and enhanced online; it harnesses the network effect of the Internet and can be very useful in reaching a large number of people rapidly. While most of previous viral marketing plans (e.g., Tupperware parties, Ford's Evil Twin Campaign and ilovebees.com) did not have any idea of the exact network, a mobile operator can construct such a network by merely considering the calls made between users. This is essential for providing an optimized list of so-called influent agents, who can have a positive effect on the diffusion speed of an ad, a new technology, a behavior, etc.

The first definitions of leaders. That study naturally led us to consider several ways to identify influent nodes in the mobile phone network; such nodes are labeled the *leaders*. Hence, we compared various *ranking vectors* of which the i^{th} entry gives the rank of node i, and the leaders correspond then to the nodes with the highest ranks. The definition of these vectors came from the sociology community (e.g., the vector of degrees), from the search engines used in the Web (e.g., the PageRank

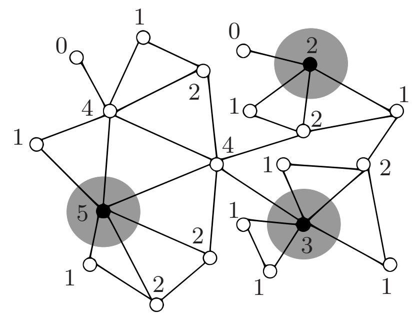


Figure 1.2: Every node has a social degree equal to the number of links between its contacts, and a social leader has a nonzero social degree greater than the one of its contacts.

vector) and finally, some of these definitions like the *social leaders* (see Fig. 1.2) or the *social degree leaders* represent our contribution in this problem. A validation test then shows that our definition of social degree leader increases by a factor two the spread of MMSs compared to the average customer and performs advantageously compared to the other definitions. A real marketing campaign was performed later on with these leaders by the company and it confirms their greater influence on the network.

In addition to the identification of leaders, the research contract allowed us to analyze the statistics of mobile phone networks in detail, which is also described in chapter 3. The cleaning of the mobile phone network gave interesting benchmarks for the stability of the relationships between customers. That collaboration led us to various descriptive statistics about mobile phone networks. This was new in the literature at that time, but is more and more analyzed in different areas these last years [12, 27, 34, 57, 84].

Analytic formula for the leaders. After the statistical study of social leaders, we derived analytic formula for the probability to be *degree leader* in a random network [11]. These leaders are closely related to the previous leaders used for the spreading of MMSs, but their definition makes their analysis tractable. We then underlined a transition phase that depends on the type of random networks we consider: in one case, the nodes of higher degree are more probably degree leader while, in the other case, their probability to be degree leader decreases. This issue is described in chapter 4. Lastly, the stability over time of the degree leader is in progress and other generalized definitions have been proposed.

Optimizing one's PageRank. Then, a second topic on the PageRank algorithm started with the collaboration of Laure Ninove [46] – and therefore that work is also present in her thesis [82]. The PageRank algorithm was already used in the leader identification problem and it gave a rank for every node proportional to the probability of presence of a *random walker* moving through the network. In our context, the walker moves from one customer to another customer via their calls. However we deviate from that context, and we rather consider the World Wide Web for which Brin and Page introduced the PageRank algorithm [85]. In that network, the nodes and the links are respectively the webpages and the hyperlinks. We analyzed the impact of hyperlinks when the owners of a set of webpages want to maximize the sum of their PageRanks. This research, stated in chapter 5, completes the studies of the maximization of one single webpage by its outgoing hyperlinks and incoming hyperlinks [6, 7, 30, 97]. That type of research allows us to analyze the sensitivity of the PageRank, often with the motivation of detecting spam among the webpages – an example of spam is webpage i that creates many artificial webpages pointing with their hyperlinks to i only to boost one's PageRank.

The possibility of negative links. The work on the PageRank algorithm draw our attention to the fact that eigenvector based techniques [59] that rank the nodes of a network, like the Hits algorithm [51] and the Salsa algorithm [63], do not allow to take into account negative links. In other words, a link from node i to node j is always considered as a positive vote, implying that it will positively contribute to the rank of j. However, there exist networks where negative links are present, like in eBay when a negative comment is exchanged between a seller and a buyer. Therefore, rather to ignore it, we propose to extend the PageRank algorithm to consider both types of links [48]. The idea, exposed in chapter 6, is then to modify the motion of the *random walker* in order to avoid the nodes that are distrusted by him. A distrusted node is a node negatively pointed to by another node that was visited before by the walker. Negative links in networks draw more and more attention in the scientific community. But their consideration in different methods is recent; let us mention, for instance, the detection of communities [100]

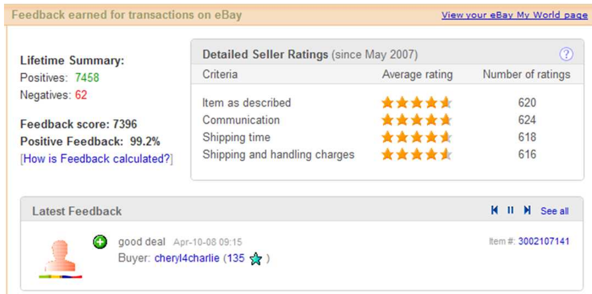


Figure 1.3: Screen shot of reputations in eBay.

and the ranking of the nodes [71].

The credibility of the links. So far, we only considered methods that assign a single rank for each node of a network. However, the rank of a node, say i, gives no information about the credibility of the links of node i. It may happen that the links of some nodes are less relevant because the raters they represent cannot be expected to be fully reliable or even honest. Chapter 7 deals with that issue by the use of *reputation systems* [47, 49] that give 2 ranks: one for the reputation of the node and one for the reliability of its links. This is crucial when we think about the increasing number of interactive ratings collected from various users on the World Wide Web: Books are evaluated on Amazon, movies are rated on Movielens, and buyers and sellers rate one another on eBay (see Fig 1.3). Therefore it is clear that Web sites have a lot to earn by promoting confidence in such interactive rating systems. Ideally, they would achieve this by penalizing raters who give random or biased ratings.

That method of reputation gives us a unique solution with the same complexity than the eigenvector based techniques like the PageRank algorithm. However, the solution is not interpretable anymore as a *random walk* moving through the network, but it minimizes some cost function. Moreover, unlike previous studies, we investigated the dynamics of reputation systems where the votes given by the raters may change over time. For a few years, that dynamical aspect of evaluations leads to new challenges, for example, the ranking of a stream of news [19].

Large Graphs and Networks. To conclude, the work of this thesis is also the result of collaborations in the group "Large Graphs and

Networks", essentially with Paul Van Dooren, Vincent Blondel, Laure Ninove, Jean-Loup Guillaume, Renaud Lambiotte, Gauthier Krings and Etienne Huens. Several ideas and advances came from discussions and seminars within this group. It also contributed to have the opportunity to obtain the research contract – from the Belgian mobile phone company – from which we derive chapter 3. That part has perhaps to be read differently from the rest since the nature of its content fits in a collaboration with the industry, where the novelty resides mainly in the type of dataset we have treated.

1.2 Chapters' presentation

We briefly expose the main problems treated in the chapters 3, 4, 5, 6 and 7. What connects these chapters is the rank or the reputation given to every node according to some method. Chapter 3 and 4 deal with a local measure of reputation defined from the neighborhood of each node, while chapter 5, 6 and 7 consider a global measure of reputation where any change in the links – deletion, variation of weight – may affect all the reputations. Moreover, chapter 6 adds the possibility of negative links and chapter 7 proposes two reputations by nodes for two separate aspects. Finally, chapters 2 and 8 respectively contain the preliminaries on networks, matrices and dynamical systems, and the conclusions on the results of this work.

Chapter 3: Leaders in mobile phone networks. Being given a list of customers with their calls – dates, durations, destinations – and their exchanges of MMSs – dates, durations – on a period of 6 months, we want to identify customers that are more influent in that mobile phone network containing millions of nodes. These customers, labeled leaders, are supposed to convince their contacts more easily than an average customer and therefore their probability to *infect* their neighbors is higher. To validate different definitions of leader, we estimate the quality of a leader by observing in the dataset the number of contacts who activate the MMS service.

Chapter 4: Degree leaders in random networks. Degree leaders are nodes whose degree is higher or equal than the degree of all of their neighbors. Such nodes may be viewed as local hubs that trigger the communication between nodes at the local level. Looking at the probability p_d of a node of degree p to be a degree leader, we search an analytic formula for random networks. In particular, we ask the question whether there is a transition for $\lim_{p\to\infty} p_d$ when the tail of the degree distribution behaves like the power-law $\sim d^{-\gamma}$, which is very general since it includes scale-free distributions (γ finite), while exponential distributions are recovered in the limit $\gamma \to \infty$.

Chapter 5: Maximization of PageRank via outlinks. The PageRank algorithm of Brin and Page introduced in 1998 is still believed to be at the hearth of the ranking provided by Google to list webpages by order of importance. It is therefore not surprising that webmasters try to boost their PageRank by choosing the best linkage strategy among their webpages. Indeed, the PageRank is completely determined by the links in the network, hence the only control of one's PageRank – disregarding creation and deletion of webpages – is on the outlinks of webpages, that is the hyperlinks. This chapter presents the optimal linkage strategy for a set of webpages that want to maximize the sum of their PageRank by only modifying its own hyperlinks.

Chapter 6: Forbidden nodes in random walks. As remarked by Massa et al. [71], the automatic robots crawling the web to build the associated network take into account every hyperlink of the webpages. Hence, the PageRank algorithm – and other eigenvector based algorithms – always interprets hyperlinks as positive opinions while some of them are used to list irrelevant webpages. However, it is possible to mention in the html code that such webpages must be ignored. We propose to go one step further, and rather to ignore them, we make the *random walker* aware of such negative opinions during his walk through the network. The difficulty remains then to keep the complexity of the new method – that extends the PageRank algorithm – reasonable for large network.

Chapter 7: Iterative filtering in voting systems. In a weighted network where the weighted links represent votes or ratings, we can believe that not all raters are trustworthy. Starting from that claim, it becomes natural to measure for every rater his credibility according to his votes and the other votes. Then the votes on one same item can be weighted according to the level of credibility of its raters, and this weighted average gives the reputation of the item. Chapter 7 proposes an iteration on the reputations and the weights of the raters: (1) the reputation are given by the weighted average of the votes; (2) the weights are function of a distance between the votes and the previous reputations. From that iteration arises problems of convergence, complexity and adaptation for dynamical votes.

1.3 Publications

We survey in this section the articles published since the beginning of the thesis.

Two research contracts. A first series of articles [13, 45, 57] have been written in the context of two research contracts, one first contract with a Belgian mobile phone company and the second with the research and development department of a French company of telecommunication. The first collaboration led to a private report that was partially made public in Heeze (Netherlands) for the Benelux meeting in 2006, and then in Grenoble (France) for the conference on *Positive Systems: Theory and Applications* the same year. The corresponding article is published in *Lecture Notes in Control and Information Sciences* and is entitled *Social Leaders in Graphs*. These results are present in chapter 3 that also contain the rest of the private report and extra discussions.

Articles not related to the thesis. The articles published during the second contract [45, 57] are not exposed in this work since they are not directly related to the original topic. The first one is published in Physica A in 2008 for the part *Statistical Mechanics and its Applications* and is entitled *Geographical dispersal of mobile communication networks*. Its analysis is also based on a Belgian mobile phone networks, but with in addition geographical home localization information. It shows that the probability that two customers are connected by a link decreases like d^{-2}, where d is the distance between the customers. It also considers the geographical extension of communication triangles and it shows that communication triangles are not only composed of geographically adjacent nodes but that they may extend over large distances.

The second article is published a bit later in *Physical Review E* in 2009 and is entitled *Role of second trials in cascades of information over networks*. In contrast with the two previous articles, it does not exploit the dataset supplied by the mobile phone company. It rather describes a theoretical cascade model where nodes are infected with probability

p_1 after their first contact with the information and with probability p_2 at all subsequent contacts. It is shown that first and subsequent trials play different roles in the propagation and that the size of the cascade depends in a non-trivial way on p_1, p_2 and on the network structure.

The rest of the articles. Chapter 4 resumes the article [11] entitled *Local leaders in random networks* and published in *Physical Review E* in 2008 with an additional discussion on upper bounds. It is a joint work with several authors with the initial goal of studying more formally the definition of social leaders introduced in [13]. It finally led to the definition of degree leaders and their properties in random networks.

The maximization of the PageRank via the outlinks presented in chapter 5 was announced in Lommel (Belgium) for the *Benelux meeting* in 2007 and then presented in Amsterdam the same year for the *ILAS conference*. The resulting article [46] entitled *Maximizing PageRank via outlinks* is published in the journal of the conference, *Linear Algebra and Its Applications*, in 2008.

The ranking of nodes in a network with positive and negative links, with the algorithms of modified *random walk* described in chapter 6, was presented in Atlanta for the *Siam International Conference in Data Mining* in 2008. The associated article [48], entitled *The PageTrust algorithm: how to rank webpages when negative links are allowed*, is published in the proceedings of that conference. An extended version of that article is currently submitted for publication.

Finally, the reputation systems in chapter 7 have been announced in the SIAM news journal [47] in 2008 entitled *Reputation systems and optimization*. An article is also accepted for the conference on *Positive Systems: Theory and Applications* for 2009, and a journal version is currently submitted for publication [49].

List of coauthors and articles. I resume here the collaborators involved in the articles mentioned above:

- Vincent Blondel,
  Professor in Applied Mathematics in the Department of Mathematical Engineering at UCL,
- Jean-Loup Guillaume,
  Assistant Professor at LIP6 (University of Paris 6),
- Julien Hendrickx,
  Postdoctoral fellow in the Laboratory for Information and Decision System at MIT,
- Etienne Huens,
  PhD in Physics from UCL,
- Gauthier Krings,
  PhD student in Applied Mathematics in the Department of Mathematical Engineering at UCL,
- Renaud Lambiotte,
  Postdoctoral fellow in the Institute for Mathematical Sciences at the Imperial College London,
- Laure Ninove,
  PhD in Applied Mathematics in the Department of Mathematical Engineering at UCL,
- Christophe Prieur,
  Assitant Professor at LIP7 (University of Paris 7),
- Zbigniew Smoreda,
  Sociologist in the Orange Labs in Paris,
- Paul Van Dooren,
  Professor in Applied Mathematics in the Department of Mathematical Engineering at U.C.L.

The articles mentioned in this section are listed here:

[11] Local leaders in random networks,
V.D. Blondel, J.L. Guillaume, J.M. Hendrickx,
C. de Kerchove and R. Lambiotte,
Physical Review E, 2008.

[13] Social Leaders in Graphs,
V.D. Blondel, **C. de Kerchove**, E. Huens,
and P. Van Dooren,
Lecture Notes in Control and Information Sciences, 2006.

[45] Role of second trials in cascades of information over networks,
C. de Kerchove, G. Krings, R. Lambiotte,
P. Van Dooren and V.D. Blondel,
Physical Review E, 2009.

[46] Maximizing PageRank via outlinks,
C. de Kerchove, L. Ninove and P. Van Dooren,
Linear Algebra and Its Applications, 2008.

[47] Reputation systems and optimization,
C. de Kerchove and P. Van Dooren,
Siam News, 2008.

[48] The PageTrust algorithm: how to rank web pages when
negative links are allowed,
C. de Kerchove and P. Van Dooren,
Proceedings of the Siam Int. Conf. in Data Mining, 2008.

[49] Iterative filtering in reputation systems,
C. de Kerchove and P. Van Dooren,
submitted to SIMAX, 2009.

[57] Geographical dispersal of mobile communication networks,
R. Lambiotte, V.D. Blondel, **C. de Kerchove**,
E. Huens, C. Prieur, Z. Smoreda and P. Van Dooren,
Physica A: Statistical Mechanics and its Applications, 2008.

Chapter 2

Preliminaries

In this chapter, we introduce background material useful for the next chapters. It firstly contains the basic definitions for the networks with some details on the random and scale-free networks. Then, we introduce nonnegative matrices that supply an important tool for the study of networks. They allow us to represent networks and to take advantage of matrix theoretic results, such as the Perron-Frobenius theorem and the power method, to treat network problems. Thirdly, we present some local and global ranking vectors that give a reputation to every node. We describe, in particular, the PageRank vector that is mentioned several times in this thesis (Chapters 3, 5 and 6). Finally, the last section is devoted to Chapter 7. It introduces fixed points of nonlinear mapping and some convergence properties.

2.1 Networks

We briefly recall in this section some basic concepts about directed networks. Then we briefly introduce random and scale-free networks; a more elaborate description of these networks can be found in [78].

Children, degree and path. A *network* $\mathcal{G} = (\mathcal{N}, \mathcal{L})$ is defined by a finite set of nodes $\mathcal{N}$ and a set of pairs $\mathcal{L} \subseteq \mathcal{N} \times \mathcal{N}$, called links[1]. Typically, we will consider $\mathcal{N} = \{1, \ldots, n\}$ and $|\mathcal{L}| = m$.

[1]In this work, we consider the terms *networks*, *nodes* and *links*, however synonyms often used are *graphs*, *vertices* and *edges*.

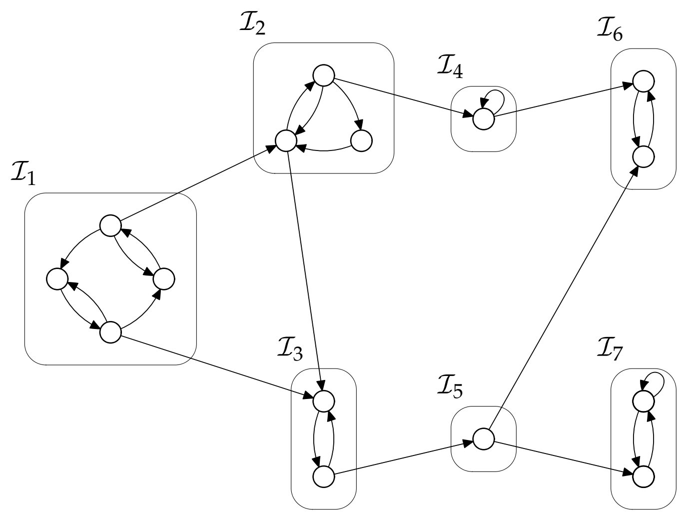


Figure 2.1: A directed network containing seven classes and among them, one initial class and two final classes.

When the set $\mathcal{L} \subseteq \mathcal{N} \times \mathcal{N}$ is considered as ordered, the links are said *directed* and $\mathcal{G}$ is called a *directed network* – otherwise, it is called an *undirected network*. The links of a network can also be weighted leading to a *weighted network*.

A directed link (i, j), represented by $i \to j$, is said to be an *outlink* for node i and an *inlink* for node j. We also say that j is a *child* of i or i a *parent* of j. The *outdegree* $\boldsymbol{d}_i$ of a node i is its number of outlinks, the *indegree* is similarly defined with inlinks.

For undirected networks, a link (i, j) is represented by $i \leftrightarrow j$. The node j is a *neighbor* of i, and d_i is then the number of neighbors of i called the *degree* of i.

A link (i, i) is called a *self-loop* on node i.

A *path* of length ℓ is a sequence of $\ell + 1$ nodes $(i_0, \dots, i_\ell)$ such that $(i_{k-1}, i_k) \in \mathcal{L}$ for $k = 1, \dots, \ell$. A *cycle* is a path that ends in its starting node, i.e., $i_0 = i_\ell$.

Accessibility and class. A node i has *access* to node j when there exists a path from i to j. If, in addition, node j has also access to node i, it is said that the two nodes *communicate* (node i communicates with j and conversely). Therefore, access and communication are equivalent for undirected networks. We will also say that a node i *has access to a set* $\mathcal{I}$ if i has access to at least one node $j \in \mathcal{I}$.

A *class* is a set of communicating nodes. We distinguish a *final class* where its nodes have no access to the nodes of other classes, and an *initial class* where its nodes cannot be accessed by nodes of other classes, see Fig. 2.1.

The network $\mathcal{G}$ is *strongly connected* if every pair of nodes in $\mathcal{N}$ communicates, equivalently the set $\mathcal{N}$ forms the unique class in the network.

A subnetwork of $\mathcal{G} = (\mathcal{N}, \mathcal{L})$ is induced by a subset of nodes $\mathcal{I} \subseteq \mathcal{N}$. Its set of links is then given by $\mathcal{L}' = \mathcal{L} \cap (\mathcal{I} \times \mathcal{I})$ that is the set of remaining links that connect the pairs of nodes in $\mathcal{I}$.

Random and scale-free networks. An important feature of networks concerns *the degree distribution* $P : \mathbb{N}_0 \to [0,1]$ that gives, for a given degree d, the proportion of nodes with degree d in the network, by construction we have

$$\sum_{d \in \mathcal{N}_0} P(d) = 1, \quad \sum_{d \in \mathcal{N}_0} dP(d) = z, \tag{2.1}$$

where $z = n/m$ is the average degree, n the total number of nodes and m the total number of links in the network. An *uncorrelated random network* is a network chosen uniformly at random from the set of all networks with a given degree distribution P. Generally, such networks exhibit no correlation between the degree of a node and the degree of its neighbors. However, in some cases, e.g., the scale-free networks defined below, several precautions need to be taken for their generation, see [18] for details. Let us mention that when the degrees of neighboring nodes are correlated, the network is said to be assortative or disassortative [77] meaning respectively that nodes are attached to nodes that have about the same degree or the converse, nodes with completely different degrees are connected.

A *scale-free network* is a network whose degree distribution follows a power law, i.e.,

$$P(d) \sim d^{-\gamma}, \tag{2.2}$$

or at least asymptotically. We observe a power law distribution in many different types of networks (social networks, information networks, biological networks, etc.). Roughly speaking, it tells us that a great part of the nodes have small degrees while a few nodes have high degrees. This scale-free property has been explained by the *preferential attachment* in the Barabási-Albert (BA) model – a new node prefers attaching to a node that already has a high degree, this implies that "rich get richer" –, this property has been measured for different networks in [40]. Typically, the parameter γ in Eq. 2.2 varies between 2 and 3, see for several examples table 3.1 in [78].

2.2 Nonnegative matrices

The representation of networks by nonnegative matrices is common and widely used for their analysis. This section focuses on elements used in this work: adjacency matrices, stochastic matrices and the Perron-Frobenius theorem. A detailed exposition can be found in [9, 36, 74, 92].

Notation. In this work, vectors are in lower case bold while matrices are in upper case. Let $\mathcal{I}$ be a subset of $\{1, \dots, n\}$. The vectors $\boldsymbol{e}_i$ and $\boldsymbol{e}_{\mathcal{I}}$ are respectively the i^{th} column of the identity matrix I and the vector with 1 in its entries $i \in \mathcal{I}$ and 0 elsewhere. The vector of all ones is denoted by $\mathbf{1}$ and the subvector $[\boldsymbol{v}_i]_{i\in\mathcal{I}}$ of a vector $\boldsymbol{v}$ is denoted by $\boldsymbol{v}_{\mathcal{I}}$.

The (square) submatrix $[M_{ij}]_{i,j\in\mathcal{I}}$ of a matrix M is denoted by $M_{\mathcal{I}}$. If $\mathcal{J} \subseteq \{1, \dots, n\}$ is another subset, then the submatrix $[M_{ij}]_{i\in\mathcal{I},j\in\mathcal{J}}$ is denoted $M_{\mathcal{I}\mathcal{J}}$.

Finally, a nonnegative matrix is a matrix in which all entries are nonnegative and a positive matrix is a nonnegative matrix with no zero entry.

Adjacency matrix. A network $\mathcal{G} = (\mathcal{N}, \mathcal{L})$ can be represented by its (square) *adjacency matrix* $[A_{ij}]_{i,j=1}^n$. The entry (i, j) of the matrix A is equal to 1 if $(i, j) \in \mathcal{L}$, and is zero otherwise, that is

$$A_{ij} = \begin{cases} 1 & \text{if } (i,j) \in \mathcal{L}, \\ 0 & \text{otherwise.} \end{cases}$$

The matrix $[S_{ij}]_{i,j=1}^n$ will be a scaled adjacency matrix, defined by

$$S_{ij} = \begin{cases} A_{ij}/d_i & \text{if } \boldsymbol{d}_i \neq 0, \\ 0 & \text{otherwise.} \end{cases} \tag{2.3}$$

When the adjacency matrix A has no zero rows, i.e., every node has at least one outlink, then the matrix S is a row stochastic matrix, i.e., a nonnegative matrix with $S\mathbf{1} = \mathbf{1}$, and we will see later on the connection between row stochastic matrices and random walks in a network.

Let us remark that a weighted network can also be represented by a *weighted adjacency matrix* $[B_{ij}]_{i,j=1}^n$. Let w_{ij} be the weight of link $(i, j) \in \mathcal{L}$, then we have

$$B_{ij} = \begin{cases} w_{ij} & \text{if } (i,j) \in \mathcal{L}, \\ 0 & \text{otherwise.} \end{cases}$$

If all weights are nonnegative, then B is a nonnegative matrix. Moreover, if it has no zero rows, then B can be scaled to obtain a row stochastic matrix P with

$$P_{ij} = \begin{cases} B_{ij} / \sum_j B_{ij} & \text{if } \sum_j B_{ij} \neq 0, \\ 0 & \text{otherwise.} \end{cases} \tag{2.4}$$

Perron-Frobenius theorem. A fundamental theorem in nonnegative matrix theory is the Perron-Frobenius theorem. It gives fundamental properties about the *spectral radius* ρ of a nonnegative matrix B, i.e.,

$$\rho(B) = \max\{|\lambda| : \exists \boldsymbol{x} \neq 0 \text{ s.t. } B\boldsymbol{x} = \lambda \boldsymbol{x}\},$$

and about its *Perron vectors* that are the left (or right) nonnegative eigenvectors associated to the eigenvalue $\rho(B)$. We will see later on how the following theorem, see also [9, 36], is useful for the definition of the PageRank vector that is the unique Perron vector of the Google matrix that we introduce further.

Theorem 1. *If $B \in \mathbb{R}^{n \times n}$ is a nonnegative matrix, then*

a) *$\rho(B)$ is an eigenvalue of B;*
b) *B has at least one Perron vector;*
c) *every positive eigenvector of B corresponds to the eigenvalue $\rho(B)$.*

Moreover, if B is irreducible, then

d) $\rho(B) > 0$;
e) B *has exactly one left (or right) Perron vector* $\boldsymbol{x}$ *that is positive and that sums to one;*
f) *every eigenvalue of maximal magnitude is an algebraically simple eigenvalue of* B*;*
g) *the eigenvalues of maximal magnitude are equally spaced on a circle centered in* 0 *and of radius* $\rho(B)$ *in the complex plane.*

Moreover, if B *is primitive, then*

h) $\rho(B)$ *is the unique eigenvalue of maximum modulus;*
i) $\lim_{t\to\infty} \left(\rho(B)^{-1}B\right)^t = \boldsymbol{y}\boldsymbol{x}^T$ *where* $\boldsymbol{y}$ *and* $\boldsymbol{x}$ *are respectively the right and left Perron vectors of* B*, scaled such that* $\boldsymbol{x}^T\boldsymbol{y} = 1$.

The properties of irreducibility and primitivity for the matrix B can interestingly be formulated in terms of networks. We say that the (directed) network $\mathcal{G}$ of a matrix B is the network represented by the adjacency matrix A, with $A_{ij} = 1$ if $B_{ij} \neq 0$ and $A_{ij} = 0$ else. Then, we have that the matrix B is irreducible if and only if $\mathcal{G}$ is strongly connected. Moreover, the matrix B is primitive if and only if $\mathcal{G}$ is *acyclic*, this means that the greatest common divisor of all cycles in the network is 1.

Stochastic matrices. We introduce now a corollary of the Perron-Frobenius theorem for the particular case of row stochastic matrices that are mainly used in this thesis.

Corollary 1. *If* $P \in \mathbb{R}^{n\times n}$ *is a row stochastic matrix, then*

a) $\rho(P) = 1$ *and is an eigenvalue of* P*;*
b) P *has at least one Perron vector.*

Moreover, if P *is irreducible, then*

c) P *has exactly one left (or right) Perron vector that is positive.*

Moreover, if P *is primitive, then*

d) $\lim_{t\to\infty} P^t = \mathbf{1}\boldsymbol{x}^T$ *with* $\mathbf{1}^T\boldsymbol{x} = 1$.

Let us finally remark that the property of irreducibility for P is not necessary to have a unique left Perron vector $\boldsymbol{x}$. When P is irreducible, its network is then strongly connected, however it is sufficient that P has exactly one final class to have a unique left Perron vector $\boldsymbol{x}$.

Proposition 1. *If P is a stochastic matrix and its network has exactly one final class, then P has a unique left Perron vector.*

For the proof, we refer to Theorem 3.1 in [91] that treats a more general case.

2.3 Ranking vectors

Important information concerning a network can be given by a *ranking vector* that provides a rank for every node in $\mathcal{N}$. The i^{th} entry of the ranking vector is then the rank of node i that represents the importance of i according to some criterion and the structure of the network. The goal of a ranking vector is to classify the nodes by order of importance in the same list, that is called *ordinal ranking*.

In this work, we are interested in finding a relevant ranking vector in three contexts. Firstly, in the context of mobile phone networks, we want to rank the customers in order to measure their influence on their contacts to apply viral marketing techniques. Secondly, in the context of the webpages of the World Wide Web, we analyze the PageRank vector of Google and we propose an extension to take into account negative links. And thirdly, in the context of (dynamical) votes, we propose two ranking vectors, one for the evaluators and another one for the evaluated items.

Flow methods and the PageRank vector. For instance, it is clear that from the Perron-Frobenius Theorem, we can extract the unique normalized left Perron vector, say $\boldsymbol{x}$ such that $\|\boldsymbol{x}\|_1 = 1$, of a nonnegative and irreducible matrix $B \in \mathbb{R}^{n\times n}$. That will provide, depending on B, a ranking vector for the network associated to B. By definition, we have

$$\rho(B)\boldsymbol{x}_j = \sum_{i\to j} \boldsymbol{x}_i B_{ij}.$$

Therefore the entry $\boldsymbol{x}_j$ can be interpreted as a rank that is a weighted sum of the ranks of its parents. That simple idea has been widely used in the literature of ranking measures where the goal is to rank every node of a network according to their positions in the structure of that network. Such methods for ranking the nodes of a network are commonly called *flow algorithms* or *eigenvector based methods* (see the survey in [59] or the book in [60]).

PageRank vector. One of the most famous flow methods, studied in this work, is given by the *PageRank algorithm* of Brin and Page (see also [10, 15, 58, 60, 85]) that is based on the Perron vector of some nonnegative matrix. Given a (directed) network $\mathcal{G}$, they consider the Google matrix G as

$$G = cS + (1-c)\mathbf{1}\boldsymbol{z}^T,$$

where $0 < c < 1$ is a *damping factor*[2], S is the scaled adjacency matrix in Eq. (2.3) and $\boldsymbol{z}$ is a positive stochastic *personalization vector*, i.e., $\boldsymbol{z}_i > 0$ for all $i = 1, \ldots, n$ and $\|\boldsymbol{z}\|_1 = 1$. Moreover, without loss of generality, we can make the assumption that each node has at least one outlink, i.e., $\boldsymbol{d}_i \neq 0$ for every $i \in \mathcal{N}$ (Refer to [10], [37] and [60] for details). Hence, it can be shown that the matrix G is row stochastic, positive, irreducible and primitive. Its unique left Perron vector is denoted $\boldsymbol{\pi}$ and, normalized to 1, that vector is labeled the PageRank vector, i.e.,

$$\begin{aligned} \boldsymbol{\pi}^T &= \boldsymbol{\pi}^T G, \\ \|\boldsymbol{\pi}\|_1 &= 1. \end{aligned} \tag{2.5}$$

The *PageRank of a node* i is then the i^{th} entry $\boldsymbol{\pi}_i$ of the PageRank vector (see [39] for details on the ordinal ranking for the PageRank).

The positivity of the personalization vector $\boldsymbol{z}$ is not necessary to guarantee the uniqueness of the PageRank vector defined in Eq. (2.5). It is sufficient to take $\boldsymbol{z}$ nonnegative, hence it can be shown that the matrix G has exactly one final class and by Proposition 1 the PageRank is unique. Actually, the uniqueness is assured because the damping factor c is strictly less than 1 [93]. This allows us to penalize some webpages by zeroing the corresponding entries of the personalization vector. For instance, such a strategy can be found in the TrustRank algorithm patented by Google [31].

We will see in the next section how the PageRank vector can be interpreted as the stationary distribution of some Markov chain representing the motion of a walker visiting at random the nodes of the network.

Local ranking vectors. The flow methods provide ranking vectors that take into account the global structure of the network. In other words, removing or adding a link may modify every entry of that vector.

[2] The parameter $\alpha = 1 - c$ called the *zapping factor* will be also used in this work.

In contrast with such methods, many local measures have been developed where the rank of a node only depends on its neighborhood in the network, e.g., its children and its parents. For example, Freeman gave a set of local ranking vectors [22, 23]. He defines the betweenness of a node i as the fraction of shortest paths between pairs of nodes that pass through the node i. Therefore, if the network represents cities connected by their roadways, a high betweenness for a city means that it is more probably crossed by dense traffic. Another simple example, in the context of social networks – where the nodes represent interacting persons –, is given by the degree d_i that is the number of contacts of the person i. Then, the persons with high degree are assumed more popular like in the Barabási-Albert (BA) model [40], and more likely to convince their neighborhood like for the spread of influence in [44].

2.4 Random walks and Markov chains

Let us imagine a walker moving in a given network $\mathcal{G} = (\mathcal{N}, \mathcal{L})$ such that he visits its nodes in $\mathcal{N}$ one by one by using the links in $\mathcal{L}$. At time $t = 0$, the walker starts in some node in $\mathcal{N}$, then, at every step, he updates his position, say node i, by choosing at random a child of i. Such a walk is called a *random walk* and the transition probabilities between the nodes are described by a transition matrix P, where P_{ij} gives the probability to visit j at the next time step being given that the walker is in node i. The simplest case is to assign the same probability to each child j of i, in that case $P_{ij} = 1/d_i$ and the transition matrix is given by the scaled adjacency matrix S.

Markov Chain. A random walk in $\mathcal{G}$ is equivalent to a *finite Markov chain* of n states in $\mathcal{N}$. Let $(X_0, X_1, \ldots)$ be a sequence of random variables with $X_i \in \mathcal{N}$, that sequence is a finite Markov chain if the transition probability at time t only depends on the state X_t, that is

$$\mathrm{Prob}(X_{t+1} = j | X_t = i_t, \ldots, X_0 = i_0) = \mathrm{Prob}(X_{t+1} = j | X_t = i_t).$$

Therefore, a random walk defined by the transition matrix P is a Markov chain, where at every time t, we have $\mathrm{Prob}(X_{t+1} = j | X_t = i_t) = P_{ij}$. We remark that P is necessarily a row stochastic matrix, and it is interesting to introduce, for $t \geq 0$, the *probability distribution vector* $\boldsymbol{x}^t$, i.e., $\boldsymbol{x}_i^t =$

$\mathrm{Prob}(X_t = i)$ for $i \in \mathcal{N}$ and $\|\boldsymbol{x}^t\|_1 = 1$, that satisfies

$$\left(\boldsymbol{x}^{t+1}\right)^T = \left(\boldsymbol{x}^t\right)^T P. \tag{2.6}$$

By Corollary 1, if P is irreducible and primitive, then $\lim_{t\to\infty} \boldsymbol{x}^t$ tends to a unique vector $\boldsymbol{x}$ that is called the *stationary distribution* vector of the Markov chain, and it satisfies $\boldsymbol{x}^T = \boldsymbol{x}^T P$. On the other hand, we know by Proposition 1 that there is a unique stationary distribution vector if the network of P has exactly one final class (we also say that the Markov Chain has a final class).

Random surfer. The formalism of Markov chain allows us to characterize the evolution of a random walker. This provides nice and intuitive interpretations for a random walker in terms of stationary distribution in the network $\mathcal{G}$. For example, the i^{th} entry of the stationary vector gives the probability to find a random walker in node i after a relatively long time, and its inverse represents the expected number of visits that a walker does before returning to node i.

In the case of the PageRank vector $\boldsymbol{\pi}$ in Eq. (2.5), the transition matrix has a unique final class, and therefore Eq. (2.6) with $P = G$ converges to the stationary distribution vector $\boldsymbol{\pi}$. The associated random walker is commonly called a *random surfer* and his motion can be described as follows: at every step, a walker, say in node i, chooses with an uniform probability a child of i. This is represented by the transition matrix S. However, rather than following a link, a walker may also decide to *zap* with a probability $1 - c$. This special motion is then defined by the personalization vector $\boldsymbol{z}$, where $\boldsymbol{z}_j$ gives the probability to visit j at the next time step. Equivalently a zapping step can be represented by the rank-1 transition matrix $\mathbf{1}\boldsymbol{z}^T$. That walk can be therefore compared with a random path that is sometimes interrupted by a zapping step where a new path is initialized. This walk mimics the motion of a hypothetic surfer who sometimes moves from one webpage to another webpage and sometimes zaps. That context explains the name of random surfer.

The stationary distribution vector $\boldsymbol{\pi}$ of the random surfer can be derived from Eq. (2.5) in the following way

$$\begin{aligned}\boldsymbol{\pi}^T &= c\boldsymbol{\pi}^T S + (1-c)\boldsymbol{z}^T, \\ &= (1-c)\boldsymbol{z}^T (I - cS)^{-1},\end{aligned}$$

where we take advantage of the invertibility of $(I - cS)$ that is strictly diagonally dominant. Since $(cS)^t \to 0$ as t tends to infinity, we have by the Taylor expansion

$$\begin{aligned}\boldsymbol{\pi}^T &= (1-c)\boldsymbol{z}^T \sum_{t=0}^{\infty}(cS)^t \\ &= (1-c)\boldsymbol{z}^T + (1-c)\sum_{t=1}^{\infty} c^t \boldsymbol{z}^T S^t, \\ \boldsymbol{\pi} &= (1-c)\boldsymbol{z} + (1-c)\sum_{t=1}^{\infty} c^t \boldsymbol{x}^t, \qquad (2.7)\end{aligned}$$

where we defined $\boldsymbol{x}^t := (S^t)^T \boldsymbol{z}$ for $t \in \mathbb{N}$. Eq. (2.7) allows to calculate $\boldsymbol{\pi}$ from the vectors $\boldsymbol{x}^t$ that have the following nice interpretation: they represent the probability distribution vector of a random surfer who never zaps ($c = 1$) and with $\boldsymbol{z}$ as initial probability distribution vector.

More generally, let $(\boldsymbol{x}^t)$ be a sequence[3] of probability distribution vectors of a random process that has a probability $(1-c)$ to be restarted with $\boldsymbol{x}^0 = \boldsymbol{z}$, that is, $\mathrm{Prob}(\boldsymbol{x}^t = \boldsymbol{z} | t \geq 1) = 1 - c$. Then, the corresponding stationary distribution vector is given by

$$(1-c)\boldsymbol{z} + (1-c)\sum_{t=1}^{\infty} c^t \boldsymbol{x}^t. \qquad (2.8)$$

2.5 Fixed points of mappings

Let $f(\boldsymbol{x})$ be a mapping from $\mathbb{R}^n$ to $\mathbb{R}^n$ and $\boldsymbol{x}^0, \boldsymbol{x}^1, \ldots$ be the sequence generated by f in the following way

$$\boldsymbol{x}^{t+1} = f(\boldsymbol{x}^t), \qquad (2.9)$$

with some initial vector $\boldsymbol{x}^0$. Let us mention that Eq. (2.9) is a *discrete dynamical system*. A *fixed point* of the mapping f is a vector $\boldsymbol{x}^* \in \mathbb{R}^n$ that satisfies

$$\boldsymbol{x}^* = f(\boldsymbol{x}^*).$$

[3]We use the notation (a_n) instead of the standard notation $\{a_n : n \in \mathbb{N}_0\}$ to distinguish sets and sequences (where the order is important).

For example, the linear mapping given by $f(\boldsymbol{x}) = \boldsymbol{x}^T G$, where G is the Google matrix, generates a sequence $(\boldsymbol{x}^t)$ that always converges to a fixed point $\boldsymbol{x}^*$. Moreover, if the initial vector is a stochastic vector $\boldsymbol{x}^0$, then we recover Eq. (2.6) and $\boldsymbol{x}^*$ is equal to the PageRank vector $\boldsymbol{\pi}$ defined in Eq. (2.5).

Energy functions. In Chapter 7, the ranking vector is the fixed point of a nonlinear mapping f. In that case, one technique to characterize the convergence of the sequence $(\boldsymbol{x}^t)$ generated by f is based on the definition of an energy function $E : \mathbb{R}^n \to \mathbb{R}$ (this is similar to the Lyapunov methods used to prove the stability of stationary points in discrete dynamical systems, see Chapter 1 to 7 of [103]). We look then for a (differentiable) energy function E that has a unique stationary point $\boldsymbol{x}^*$ in a subset $\mathcal{U} \subseteq \mathbb{R}^n$, such that $E(\boldsymbol{x}^*) = 0$,

$$\begin{aligned} 0 < E(\boldsymbol{x}) \qquad & \text{for all } \boldsymbol{x} \in \mathcal{U} \setminus \{\boldsymbol{x}^*\}, \\ E(\boldsymbol{x}^{t+1}) < E(\boldsymbol{x}^t) \qquad & \text{for all } \boldsymbol{x}^t \in \mathcal{U} \setminus \{\boldsymbol{x}^*\}. \end{aligned}$$

These assumptions guarantee the convergence of the sequence $(\boldsymbol{x}^t)$ to the stationary point $\boldsymbol{x}^*$ if $\boldsymbol{x}^0 \in \mathcal{U}$.

Contraction mappings Another technique to prove the convergence of Eq. (2.9) is based on the contraction property of a mapping. We say that f is a *contraction mapping* in a subset $\mathcal{U} \subseteq \mathbb{R}^n$ if for all $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{U}$ with $\boldsymbol{x} \neq \boldsymbol{y}$, we have

$$\|f(\boldsymbol{x}) - f(\boldsymbol{y})\| < \|\boldsymbol{x} - \boldsymbol{y}\|,$$

where $\|.\|$ can be any operator norm. It follows by the *Banach fixed point theorem* that f has a unique fixed point $\boldsymbol{x}^*$ in $\mathcal{U}$ and that every sequence in Eq. (2.9) with $\boldsymbol{x}^0 \in \mathcal{U}$ tends to $\boldsymbol{x}^*$ (see [28] for a general presentation of fixed point theory).

We could be interested in the greatest subset $\mathcal{U}^* \subseteq \mathbb{R}^n$ such that for all $\boldsymbol{x}^0 \in \mathcal{U}^*$, the sequence $(\boldsymbol{x}^t)$ generated by Eq. (2.9) converges to a unique fixed point $\boldsymbol{x}^*$. That subset $\mathcal{U}^*$ is called the *basin of attraction* of $\boldsymbol{x}^*$ and the dynamical system is said *globally convergent* in $\mathcal{U}^*$.

The dynamical system in Eq. (2.9) is said to be *locally convergent* if there exists a neighborhood $\mathcal{U}$ of $\boldsymbol{x}^*$ such that the sequence $(\boldsymbol{x}^t)$ converges to $\boldsymbol{x}^*$ for all $\boldsymbol{x}^0 \in \mathcal{U}$. If f is differentiable, a sufficient condition is given by

$$\|\nabla_{\boldsymbol{x}} f(\boldsymbol{x}^*)\| < 1$$

for some operator norm $\|.\|$. This is equivalent to having a local contraction for the mapping f in a neighborhood of $\boldsymbol{x}^*$.

Rate of convergence. Let $(\boldsymbol{x}^t)$ be a sequence generated by Eq. (2.9) that converges to $\boldsymbol{x}^* \in \mathbb{R}^n$ and $\boldsymbol{\epsilon}^t$ be the error vector at time t defined by

$$\boldsymbol{\epsilon}^t = \boldsymbol{x}^* - \boldsymbol{x}^t.$$

We say that the convergence of the sequence $\boldsymbol{x}^t$ is *q-linear* if

$$\sup \lim_{t \to \infty} \frac{\|\boldsymbol{\epsilon}^t\|}{\|\boldsymbol{\epsilon}^{t+1}\|} = k < 1,$$

for a constant $k \neq 0$ that is called the *rate of convergence* and some operator norm $\|.\|$. For example, the sequence generated by the mapping $f(\boldsymbol{x}) = \boldsymbol{x}^T G$ and a stochastic initial vector $\boldsymbol{x}^0$ q-linearly converges to the PageRank vector $\boldsymbol{\pi}$ and its rate of convergence k is upper bounded by the damping factor c (see [20, 21, 33]).

Chapter 3

Leaders in Mobile Phone Networks

In this chapter, we study a large network supplied by a Belgian mobile phone company where several millions of customers are connected to each other according to their calls. The work is the result of a successful collaboration with the industry where we deal with **three different issues**: (1) cleaning the data set to remove uncorrect and irregular connections, (2) determine the most influent customer able to initiate a viral marketing campaign and (3) visualizing the mobile phone network by reducing its size.

The **cleaning step** clearly shows that most of the connections are not stable over time, this claim is confirmed by the novel concept of *connection stability* that measures the regularity over time of every link in the network.

The **viral marketing issue** leads to the identification of several types of leaders that are susceptible to initiate a viral marketing campaign. The definitions of *social degree leader* and *social leader* show good results in our validation test. Moreover a real viral marketing campaign based on these definitions and handled by the same company has shown a significant lift in the spread of specific products through the customers.

Finally, another application of social leader remains in the **visualization of a large network**. They allow to reduce the size of a network by aggregating its nodes around these local leaders, and therefore it decreases the complexity of existing algorithms to visualize a network.

3.1 Introduction

Motivation. We define the concepts of social degree and social leader in the context of social networks. We remind that such networks are made of *social* links that find their origin in social interactions between the nodes. Well studied examples are emails networks, webpages networks or co-authors networks that are presented among other types of networks in [78]. The definitions of social degree and social leader give local centrality measures for every node in a network. In some way, they extend the basic measures of degree and clustering coefficient of a node, but their definitions remain simple: the social degree of a node, say i, is the number of cycles of length 3 starting in i, and a social leader is a node that has a social degree greater than its neighbors, see Fig. 3.1. In order to validate such definitions, we motivate their introduction to identify influent customers in social networks (word of mouth effect), and visualize large networks. This is essential for viral marketing where we need to identify agents that are influent in the network in order to speed up the word of mouth effect. Influent customers should not only be well connected with many contacts but their contacts should also have a large probability to know each other. These considerations lead us to the concept of social degree and social leader. Therefore, we extract the information from the network linking the entities rather than using the local data available for each of them (language, locality, etc.). The motivation for mainly focussing on the linkage between the entities to apply data mining techniques can be found in [35, 68].

In addition to these new concepts, we experiment network mining tools for the cleaning of the mobile phone network supplied by the Belgian company. This is essential to eliminate the noise created by those calls that do not represent any social interaction, e.g., commercial calls, and to quantify some global statistics. It clearly gives an interesting benchmark for mobile phone networks that are still under investigation in the literature [12, 57, 84].

Definitions. Many other local measures exist. They are often called measures of centrality because they characterize how central the nodes are in the network. In social networks, for example, Freeman gave a set of measures of centrality where some of them are based on betweenness [22, 23]. Generally betweenness of a node i is the fraction of shortest paths between node pairs that pass through the node i. Another simple

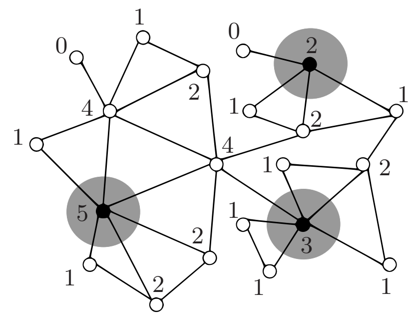


Figure 3.1: A social degrees distribution with three social leaders.

measure of centrality, already mentioned above, is given by the degree of node i. Social leaders are somehow related with these last two kinds of central nodes. Their role is to connect people as explained by Newman in the section *Clustering coefficients* in [79]. We will see in the sequel how the social degree and social leaders give interesting results in viral marketing and in aggregation of large networks for visualization. We define these two concepts more formally.

Definition 1. The social degree of node i in a directed network $\mathcal{G}$ is the number of cycles of length 3 starting in i. If $\mathcal{G}$ is undirected, we divide by two that number to not consider both directions of cycles.

Definition 2. A social leader in a network $\mathcal{G}$ – directed or not – has a social degree greater than his neighbors and different from zero.

The sum of all social degrees is exactly equal to three times the number of *triangles* in the network, i.e., the number of cycles of length 3 (counted without considering both directions when the network is undirected). This number is high for social networks since we have transitivity of friendship, *the friends of my friends are my friends*. A first condition to be a social leader is to have a social degree different from 0, in other words to belong at least to a trio of friends. The second condition is based on a local maximum: a social leader has a social degree greater or equal to the social degrees of his neighbors, see Fig. 3.1. Two social leaders cannot be neighbors except when they have the same social degree. A typical example is a clique: every node is connected to every node and then all nodes have the same social degree.

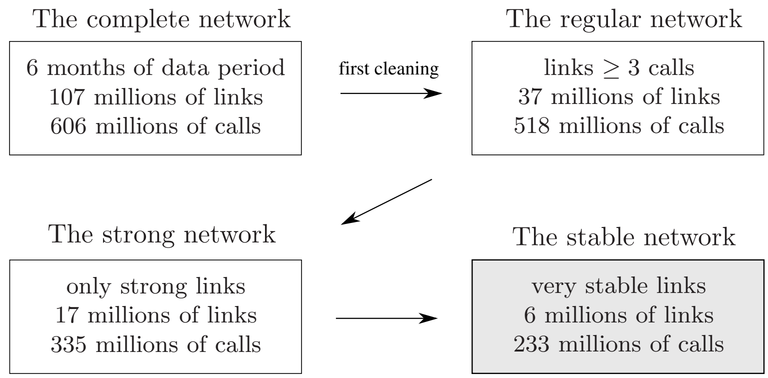


Figure 3.2: Noise cleaning steps: from the complete network to the stable network.

The aggregation of a network around the social leaders is discussed in the section of visualization of the network. The idea is to look at the subnetwork induced by the subset of nodes represented by the social leaders. The links (i, j) of that subnetwork are the path of length at most 3 in the original network from the social leaders i to the social leader j.

Social leaders can also be viewed as an optimal meeting point for their neighborhood and have interesting applications. Moreover the calculation of social degrees and social leaders has a linear complexity $O(m \cdot d_{max})$, where m is the number of links and d_{max} the largest degree in the network (see [61] for details on the counting of triangles in large and sparse networks).

Structure. The next three sections describe the data set of a mobile phone company and the different investigations we made. The first one consists in extracting and cleaning the data set to obtain several exploitable mobile phone networks. Then, in the two next sections, we look at two particular problems: the identification of leaders and the visualization of such networks. The conclusion, in the last section, gives insights on the possible further research.

3.2 Presentation of the data

Mobile phone networks are typical examples of social networks. They supply useful benchmarks for viral marketing. In order to consider the problem of identifying customers, i.e., nodes in the networks, that play an essential role in the word of mouth effect, it is necessary to prepare the network by cleaning it. In the sequel, the company that supplied the data set will be labeled the **Company**, and the group of other concurrent companies will be labeled the **Other**. Two types of cleaning have been done on the data: data cleaning and noise cleaning.

Data cleaning. With data cleaning, we have removed double lines and contradictory information about a single customer. After most of the data cleaning has been carried out, there are still a few double entries remaining and possibly other error coming from the record itself of the calls. However, the results given in the sequel are not sensitive to these remaining double entries since they represent a negligible proportion of the calls.

Before applying the noise cleaning step on the data, we focus on the network obtained by aggregating data over the total period of 6 months. Every node of that network is a MSISDN that uniquely identifies a subscription in the mobile phone. Therefore we can reasonably assume that it represents exactly one customer. Every link (i, j) has a weight corresponding to the total number of calls made during the 6 months period from node i to node j. An alternative is to replace the total number of calls by the total duration from node i to node j, but the correlation between the total number of calls and the total duration is high. Therefore the relative weights between links hardly change when applying this alternative. The network contains around 8 millions of nodes, 107 millions of directed links and 606 millions of calls. It is labeled the *complete network* and is at the top of Fig. 3.2. From that complete network, several choices were made to simplify the network. After the remark on hidden links in the next paragraph, three sorts of noise cleaning are applied on the complete network and they lead to the *regular network*, the *strong network* and finally, the *stable network*. Each of them amounts to remove a subset of nodes and a subset of links from the complete network. The stable network gives the smallest network used for the viral marketing issue in the next section.

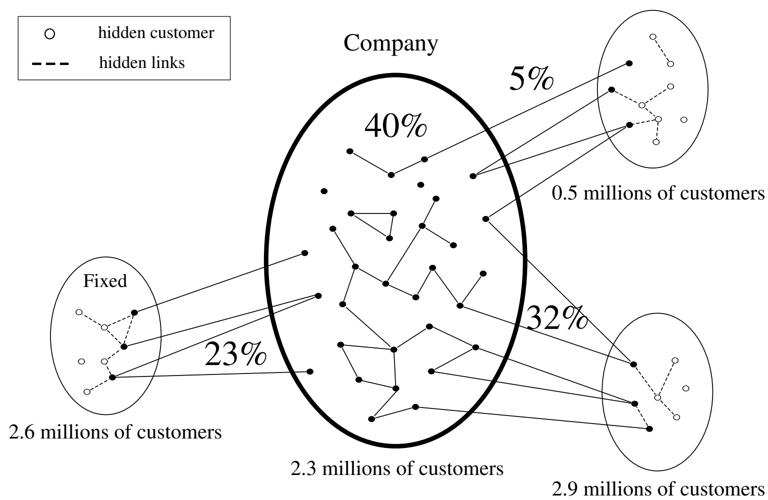


Figure 3.3: A month of data contains 40% of links Company-Company and a total of 60% of links Company-Other with among them 23% of links concerning the fixed phones.

A difficulty concerning the complete network comes from its hidden links between customers that are not in the Company. Indeed, the calls of such customers are not in the data set. Only calls with at least one customer in the Company are registered. For instance, the aggregation over a month among the 6 months of the period is schematically pictured in Fig. 3.3. Only the calls having at least one customer in the Company are visible: we have about 137 millions of calls, 25 millions of links and 8.15 millions of nodes. Taking into account only the customers in the Company, we obtain a new network with 71 millions of calls, 10 millions of links and 2.3 millions of nodes. The difficulty about hidden calls is avoided later on by focusing only on calls between customers of the Company.

Noise cleaning has eliminated "singular" customers such as automatically generated calls by machines, and customers with very few calls. To decide whether a link is socially poor or not, we use the following threshold: the number of calls through the link during the 6 months period must be at least 3. In that way, the links with one or two calls are removed. Such links represent 2/3 of the links in the complete network. However in term of call, they only represent 15% of the total number

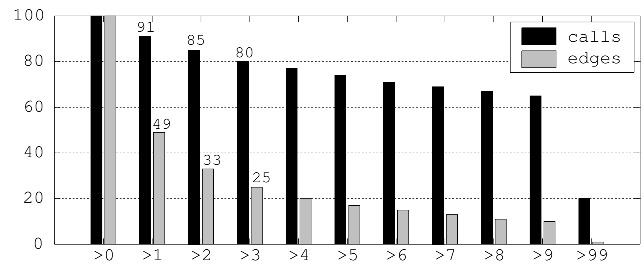


Figure 3.4: Percentage of remaining calls and links for links having strictly more than 0,1,...,99 calls.

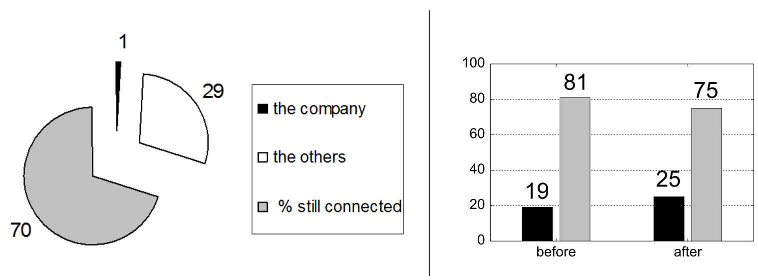


Figure 3.5: Removing the links with strictly less than 3 calls creates 30% of passive nodes, but only 1% of these nodes belong to the Company. Therefore, after removing the passive nodes the proportion of Company nodes in the network is 25%.

of calls, see Fig. 3.4. These links are assumed to be occasional contacts without stability over time and are therefore removed.

That step removes links and by consequence it creates new “passive nodes”, i.e., nodes without a link. The pie chart in Fig. 3.5 shows that 30% of the vertices become passive after noise cleaning. Only 1% among them are in the Company while 29% among them are in the Other. After the cleaning, we then have a greater proportion of Company nodes: it goes from 19% to 25%. This first cleaning (that removes poor links and passive nodes) yields the regular network shown at the top right of Fig. 3.2.

From that network, it is interesting to look at the distribution of the nodes according to the calls they give and receive, and according to their company. We can distinguish three classes of nodes: the receivers (R)

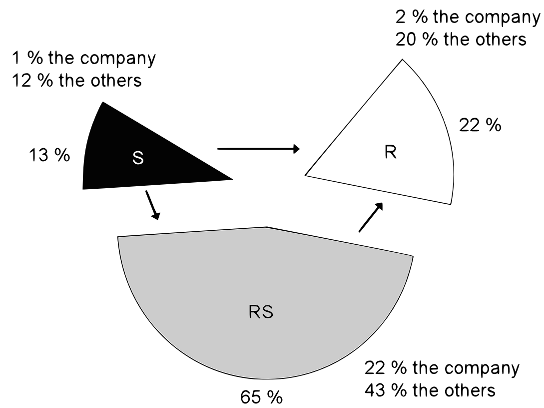


Figure 3.6: Distribution of the customers according to the three classes: Receivers (R), Senders (S) and Receiver-Senders (RS). For each class, we have two possibilities: in the Company or in the Other.

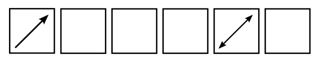
that only receive calls, the senders (S) that only call, and the others (RS) that receive calls and call. In Fig. 3.6, we sum up the percentage of nodes from the Company and obtain the 25%. The great majority (88%) of Company nodes receive and give calls. This is less true for Other nodes (57%), however this is due to the hidden links and the hidden nodes, see Fig.3.3. In other words, a customer in Other and in the class S may have received calls that was merely not registered in the data set.

A further cleaning step of the network is to identify the weak links and the strong links. Weak links (i, j) are links for which there is no link (j, i), meaning that the calls through the link go in only one direction. The other links are labeled strong links. In the regular network, we have 56% of weak links and 44% of strong links, but the strong links represent 2/3 of the calls. The proportion of strong nodes, that we define as nodes having at least one strong link, is 40%. Therefore, keeping only the strong removes 60% of the passive nodes. The resulting network is labeled the strong network and is shown at the second leaf of the tree in Fig. 3.2.

The last cleaning step of the network takes into account the regularity of the calls between two customers over the 6 months, where the period is split in 6 parts (of one month each). Then we define the connection stability of a link by looking at the nature of the link for every month (weak or strong).

Definition 3. The connection stability (CS) of a link (i, j) is a pair of values (w, s), where w gives the number of months during which there is at least one call between i and j, and where s gives the number of months during which i calls j and conversely.

For example, we can represent the calls through the link (i, j) by 6 boxes, each box corresponding to a month. Then, a box is empty if there is no call that month, or it contains a weak link or a strong link depending on the calls between i and j. Therefore, we have for the following boxes:



a connection stability equal to $(2, 1)$ (2 active months and 1 strong link during one month).

The distributions of links and calls for the connection stabilities are given in the two tables of Fig. 3.7. We observe again the same tendency seen in the previous filtering: the relative proportion of calls increase when we consider "more stable" links. For instance, only 5% of the links have a connection stability of $(6, 6)$, but they contain 28% of the calls of the regular network. This gives us another filtering method to remove weaker links from the regular network. By keeping only links that have a connection stability (w, s) with $s \geq 4$, we obtain the stable network represented at the third leaf of the tree in Fig. 3.2. The percentages of these "very stable" links are in bold in the tables of Fig.3.7, they represent 15% of the links and 46% of the calls of the regular network.

3.3 Viral marketing

The goal of this section is to identify leaders in the network in order to apply some particular viral marketing techniques. Therefore the leaders are customers that better propagate a product or an information through the network. The measure of influence for the leaders is based on the stable network with very stable links described at the end of the previous section, see Fig. 3.2. The method of validation compares the efficiency of several types of leaders in the context of the adoption of a mobile phone service (the usage of Multimedia Messaging Service).

Validation model. The comparison method for the leaders uses dynamical data for MMS. We know origins, destinations and dates of

CS	0	1	2	3	4	5	6	Total
1	13	1						14
2	18	2	2					22
3	13	2	3	1				19
4	6	2	3	2	**1**			15
5	3	1	2	3	**2**	**1**		12
6	2	1/2	2	2	**3**	**3**	**5**	18

(a)

CS	0	1	2	3	4	5	6	Total
1	4	1						5
2	6	1	1					8
3	6	1	2	1				10
4	4	1	2	2	**1**			10
5	4	1	2	2	**2**	**2**		13
6	8	1	2	3	**5**	**8**	**28**	54

(b)

Figure 3.7: Percentage of links (a) and calls (b) according to the Connection Stability (CS), in bold the percentages of links and calls in the stable network.

all MMSs sent during the 6 months period. The idea is to focus on customers who receive MMSs during the first 3 months without sending MMSs, and then who send their first MMS during the last 3 months. It is assumed that those customers were not yet using the MMS technology, and that they adopted this service thanks to someone who sent him an MMS during the first three months. Since MMSs between two customers not in the Company (in Other) are not in the data, the only customers for whom we are sure they sent their first MMS the last three months are customers in the Company (they are about 2.4 millions in the stable network). Therefore, we only consider these customers.

During the first 3 months, 64 673 nodes received MMSs without sending MMSs; they are called the **Receivers** in the rest of this section. Among them, 6 358 customers sent their first MMS during the last 3 months; they are called the **Followers** in the rest of this section. We already remark that 10% of the Receivers become infected while only 1% becomes infected if they are not encouraged by MMS senders (spontaneous emission of MMSs).

The different leaders. Clearly the exact influence between two customers is difficult to measure. In the social network literature, leaders are identified as central nodes. Centrality is often related to nodes that make the network stable. Depending on the context, removing such central nodes may lead to a disconnected network, may stop the spreading of a disease, etc. We define 8 different types of leaders that are taken using the stable network with the 6 months period:

- **Social degree leaders**: customers with the highest social degrees, see Def. 1.
- **Social leaders**: customers with a social degree greater than his neighbors and different from zero.
- **PageRank leaders**: customers with the highest PageRanks using Eq. (2.5) with $c = .15$ and the stochastic matrix P defined in Eq. (2.4) where the weights w_{ij} are the number of calls from i to j.
- **Outdegree leaders**: customers with the highest numbers of contacts.
- **High value leaders**: customer with the highest numbers of given and received calls.
- **Total MMS leaders**: customers with the highest numbers of MMSs sent during the first 3 months.
- **Call flow leaders**: customers with the highest numbers of calls given to a Receiver.
- **MMS Flow leaders**: customers with the highest numbers of MMSs sent to a Receiver.

All definitions of leaders, except the one of social leader, depend on the number of leaders we want to have. For example, in order to have k PageRank leaders, we choose the k customers with the highest PageRank.

Let see the motivations for each definition. The first two definitions concern our approach for the identification of leaders. The PageRank leaders depend on a global measure and correspond to the most visited

Type of Leaders	Receivers	Followers	% Followers
All	64 673	6 358	9.8
Social Degree	1517	304	20
Call Flow	1206	242	20
High Value	1205	236	19.6
Out-Degree	1222	231	18.9
PageRank	1227	207	16.9
Social Leader	1205	197	16.4
MMS Flow	1259	192	15.3
Total MMS	1205	100	8.3

Figure 3.8: For every type of leader, their corresponding numbers of Receivers, then the numbers of Followers and the percentage of infection.

customers if some random walk is applied on the network where the links are weighted according to the number of calls. The outdegree leaders, the high value leaders and the total MMS leaders are the most active customers in terms of number of contacts, number of calls or number of MMSs. Finally the call flow leaders and the MMS flow leaders are respectively related to the the high value leaders and the total MMS leaders, but the number of calls or MMSs is maximal for a single link that points to a Receiver.

Results. The table in Fig. 3.8 shows the results. The social leaders in that table are defined slightly differently: in addition to be social leader, their social degree and their outdegree must be at least equal to 3. These leaders sent MMSs to 1205 Receivers. Since there are 754 such social leaders, we consider the 754 highest social degree leaders, call flow leaders, etc. We remark that the first 3 types of leader are two times more successful in infecting their community with the MMS technology than the average customer (All). On the other hand, the total MMS leaders have a smaller efficiency than the average customer. That can be explained by the fact that these Receivers receive many MMSs without sending any MMS. Therefore, they seem to reject that technology the first 3 months and they will also not adopt it the last 3 months.

It is interesting to remark in Fig. 3.9 the small overlap between social degree leaders and call flow leaders that makes these two types of leaders complementary. By mixing them the percentage of leaders increases until

	SD	CF	HV	OD	PR	SL	MF	TM
Social Degree	100							
Call Flow	7	100						
High Value	27	18	100					
Out-Degree	20	6	47	100				
PageRank	25	19	61	32	100			
Social Leader	34	7	15	9	16	100		
MMS Flow	1	5	2	1	1.6	1	100	
Total MMS	.2	.5	1	.4	1.5	1	12	100

Figure 3.9: Overlap in percentage between the 754 items with highest values.

22.1% (1269 Receivers and 280 Followers). As expected, the greater intersection with the social degree leaders comes from the social leaders. There is also a large intersection between the outdegree leaders, the PageRank leaders and the high value leaders. For example, PageRank leaders contain 61% of high value leaders in its set. On the contrary, the MMS flow leaders and total MMS leaders are largely different from the other type of leaders, but as already said, their influence on the Receivers is poor.

3.4 Visualizing a network

Several algorithms for visualizing large networks consider a repulsive force between every pair of nodes and an attractive force between every pair of connected nodes. Then they calculate the equilibrium of these forces and project this in the plane [32, 41]. These methods provide good results, but are expensive for networks with millions of nodes as the mobile phone networks in the tree of Fig. 3.2. The following visualization was made on another time period than previously studied, but with the same meaning concerning the links and the nodes. That network has about 2 millions of nodes and 10 millions of links, and in order to visualize it, we aggregated the network around its social leaders for several recursive levels until reaching a network of about 60 000 nodes.

An aggregation step from the network $\mathcal{G}_0$ to the aggregated network $\mathcal{G}_1 = (\mathcal{N}_1, \mathcal{L}_1)$ is as follow: the set of nodes $\mathcal{N}_1$ contains the social leaders of $\mathcal{G}_0$ and a (directed) link (i, j) belongs to $\mathcal{L}_1$ when there exists a path

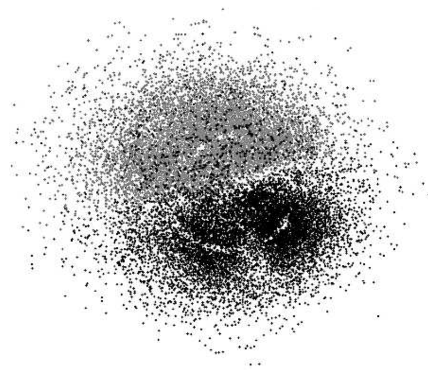


Figure 3.10: After aggregation, visualization of the customers that have been registered in a locality of Flanders (grey) or in a locality of Wallonia (black).

of length at most 3 from i to j in the network $\mathcal{G}_0$.

From that reduced network, we use the algorithm in [32] that calculates the equilibrium of the forces in the network. That equilibrium minimizes an energy function depending on the attractive and repulsive forces between nodes. Fig. 3.10 shows that visualization with customers registered in two different types of localities: Flanders and Wallonia (Brussels has been removed for the visualization). Clearly, the addition of weaker neighbors around their respective social leaders will hardly change that energy function. This is due to the fact that each non social leader node is attracted by its own social leader i and often by other neighbors of the node i thanks to the triangles around the node i. Moreover, the geographical position of a node is in most cases very close to the one of its closest social leaders, see [57].

3.5 Conclusions

Results. Among the numerous possibilities of centrality measures for the nodes of a network, the new concept of social leaders seems appealing because of its intuitive aspects, its computability and its surprising results in real data sets. Firstly, social leaders can be considered as influent persons in the network, since they are able to initiate the spreading of products. Secondly, taken as aggregation point, they allow to reduce successively the number of nodes in the network.

The definition of social degree and social leaders, see Def.1,2, is easy to extend to directed weighted networks. A general definition for the social degree of a node comes by considering the weighted adjacency matrix B, with B_{ij} equals to the weight of the link (i,j) and no self loop, i.e., $B_{ii} = 0$ for $i = 1, \ldots, n$:

Definition 4. The social degree of node i in a directed and weighted network is the i^{th} entry of the vector $\operatorname{diag}\left(B^3\right)$.

The general definition of social leader does not change; it remains based on a local maximum according to the general definition of the social degree of a node.

Additional constraints allow us to consider a subset of leaders among the social leaders as already done in the viral marketing section: the social degree and the degree of the social leader had to be greater than 3. That generated a subset of leaders that are more effective in spreading the service. Another variant comes when considering the social leaders with social degrees very close to the social degrees of their neighbors. Such leaders can be dethroned easily by small perturbations on the links or the nodes. Therefore, we can require that a social leader has a social degree at least r times greater than the social degrees of its neighbors. For $r \leq 1$ we will on the contrary relax the initial constraint and increase the number of social leaders. A last way to generalize the definition of social leader is to consider cycles of length different from 3. For example, the leaders for the cycles of length 2, studied in the next chapter, are then the node with a larger degree than that of its neighbors (for undirected and unweighted networks).

Future research. The temporal stability of the social leaders is important when we consider dynamical networks. In our mobile phone network, a social leader for the first 3 months remains not necessarily a social leader for the next 3 months. The study of temporal stability of links in a mobile phone network has been studied in [34], and the stability for different types of leaders in a mobile phone network is under investigation. The dynamical aspect of a mobile phone network is twofold: customers may change of company, mobile phone, contract,

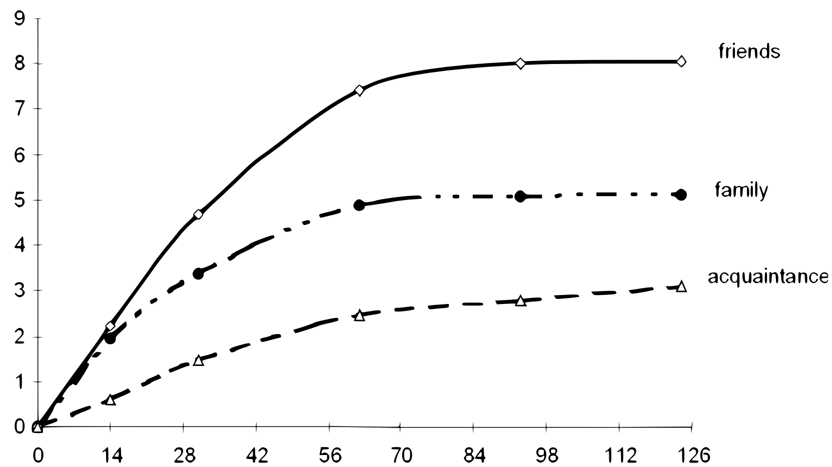


Figure 3.11: Saturation of cumulated contacts as a function of time.

etc. and the structure of the network varies with the calls. In order to deal with dynamical networks, we have used time windows of length L on the time-varying mobile phone network (L was equal to 3 months). Within each time window, data were aggregated, with which we mean that we consider the network of all links made during that time window (weights were associated with each link in order to reflect the connection time during that window). The next time window moved over a fixed period of time Δ (Δ was fixed to 3 months). Generally speaking, two consecutive time windows are allowed to overlap. In this manner we generated a new discrete set of aggregated networks. The length L of the sliding window and the time shift Δ have to be chosen judiciously in each application. For example, we know that the cumulative function of contacts over time for any node saturates after a while. Indeed, in [94] Smoreda observed 312 families in 1998 during 4 months. After 3 months, the newly called people are negligible, see Fig. 3.11. Such a monotone increasing function is often approximated by $M(1 - e^{-\lambda t})$, which corresponds to a model where a person has a total of M acquaintances and at each time contacts a random person of that population with equal probability (the total population is reached only after infinite time and λ indicates how fast the function approaches M).

Moreover, the time dependent data obtained from social networks (like calls and emails to friends or colleagues) often have a periodic behavior. For example, the MMS traffic of an individual has a typical 24 hours pattern. If one takes a time shift Δ that is a multiple of this period, the cyclic behavior will filter out the periodicity due to the day/night activity. One could wish to eliminate in a similar fashion the periodicity due to the weekday/weekend activity by taking a time shift of one week, etc.

All these remarks on the stability of the social leaders and the alternatives for defining the dynamical network lead us to carefully interpret the results of efficiency in Fig. 3.8. A particular point deserves some attention. Indeed, some leaders (like the social leaders) imply that Followers received MMSs not only from them but also from other customers. Actually, a great portion of Receivers receive exactly one MMS, see Fig. 3.12(a), and therefore the influence comes principally from the leader himself. Nevertheless, the number of Senders clearly increases the proportion of Followers, see Fig. 3.12(b). This observation has to be crossed with similar results for other services [65, 98] where the phenomenon of pressure is observed and discussed.

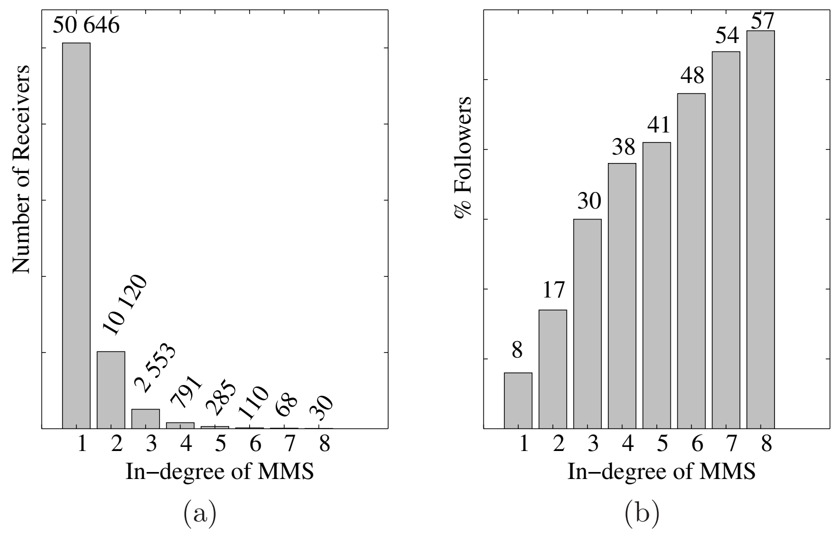


Figure 3.12: (a) Distribution of Receivers according to the number of Senders; (b) percentage of Followers among the Receivers with a given number of Senders. For example, 10 120 Receivers receive MMSs from two different Senders and 17% among them became active the three last months.

Chapter 4

Degree Leaders in Random Networks

We focus on local leaders, closely **related to the social leaders** introduced in the previous chapter, that we label *degree leaders*. They are also based on a local maximum, indeed, their degrees are higher or equal than the degrees of all of their neighbors. Therefore, they can be viewed as social leaders when we take into account the cycles of length 2 instead of 3.

This definition has the advantage to make possible a **theoretical analysis** for the degree leaders in random networks. An analytical expression is found for the probability of a node of degree d to be a degree leader. The difficulty with cycles of length 3 or greater comes from their weak proportion in random sparse networks. It is true that some models allow to reproduce the proportion of these cycles observed in real social networks, but then a theoretical approach becomes too complicated.

This quantity is shown to exhibit a **transition** from a *rich gets richer* to a *rich is poor* situation when the tail of the degree distribution behaves like the power-law $\sim d^{-\gamma}$ with $\gamma = 3$. Theoretical results are verified by computer simulations, the importance of finite-size effects is discussed and upper bounds on the proportion of degree leaders are given.

4.1 Introduction

Motivation. It is now well-known that degree heterogeneity [14, 95] and, especially the presence of hubs, are important factors that may radically alter the propagation of *data*, e.g., rumors [65], opinions [24, 55] or a virus [17], in a network and may provoke its weakness in front of targeted attacks [72, 86].

The fundamental role played by hubs in the above processes has therefore motivated a detailed study of the extremal properties of networks. Different works [53, 75] have focused on the properties of the degree of the leader, i.e., the node with the highest degree, on the probability that the leader never changes and on related leadership statistics [67]. These approaches, based on the theory of extreme statistics [25], have provided an excellent description of the behavior of the global extrema in the network but, surprisingly, the statistics of local extrema has not been considered yet.

There are several reasons, though, to focus on *local leaders*, namely nodes whose degree is larger or equal to the degree of their neighbors and on *strict leaders*, namely nodes whose degree is strictly larger than the degree of their neighbors (see Fig. 4.1). Such nodes may be viewed as local hubs that trigger the communication between nodes at the local level. Indeed, individuals usually compare their *state* (e.g., opinion, wealth, idea, etc.) with the *state* of their direct neighbors, thereby suggesting that a local leader might have a preponderant role in its own neighborhood, whatever the absolute value of its connectivity. As a rich among the poor, a local leader might therefore have a more dominant role than as a rich among the richest. From a marketing point of view, for instance, the identification of such nodes might be of interest in order to target nodes that play an important role within *circles of friends* [13]. Let us also stress that local leaders form a subset of nodes that might grasp important characteristics of the whole network and could be helpful in order to visualize its internal features.

Structure. In this chapter, we focus on the properties of local leaders in uncorrelated random networks, i.e., networks where the degrees of neighboring nodes are not correlated [77]. In section 4.2, we derive an analytical formula for the probability p_d for a node of degree d to be a degree leader (p_d can also be interpreted as the proportion of degree leaders with degree d in the network). We show that this probability

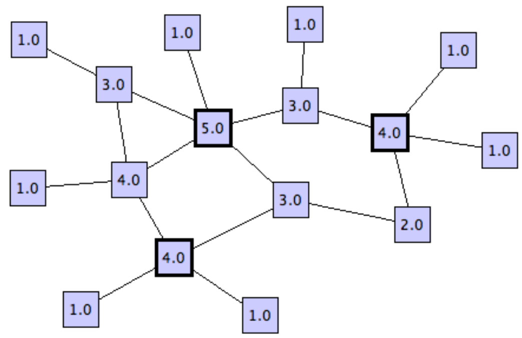


Figure 4.1: Sketch of a random network composed of 16 nodes. The network possesses 3 local leaders, two of them being strict leaders.

undergoes a phase transition where the control parameter is the degree distribution itself [96]. When the tail of the distribution decreases faster than a power-law $\sim d^{-\gamma}$ with $\gamma = 3$, the probability to be a local leader goes to one for large enough values of d. When the tail of the distribution decreases slower than $\sim d^{-\gamma}$, in contrast, this probability vanishes for large enough degrees. In section 4.3, we verify our theoretical predictions by computer simulations and show how finite size effects may affect the above transition. In section 4.4, we derive several upper bounds on the probability p to be a degree leader in a random network. In section 4.5, finally, we conclude and propose generalizations of the concept of local leader.

4.2 Being rich among the poor

Probability to be degree leader. Let us consider an undirected random network determined by its degree distribution $P(d)$, i.e., the proportion of nodes with degree d. Such a network is chosen uniformly at random from the set of all networks with a given degree distribution P (see Chapter 2, Section 2.1).

We will assume that there are no nodes with degree $d = 0$, which is reasonable as such nodes are excluded from the network structure.

Hence, the average degree is

$$z = \sum_{d=1}^{\infty} dP(d).$$

Let us now evaluate the probability p_d for a node of degree d to be a degree leader – the case of strict leaders will be discussed at the end of this section.

Proposition 2. *Let $\mathcal{G}$ be a random network with degree distribution $P(d)$, then probability p_d for a node of degree d to be a degree leader is*

$$p_d = \left(\frac{\sum_{j=1}^{d} jP(j)}{\sum_{j=1}^{\infty} jP(j)} \right)^d . \tag{4.1}$$

Proof: One first has to look at the probability q_j that a neighbor of the node under consideration has a degree j. In a network where the degrees of adjacent nodes are statistically independent, the probability q_j is given by the proportion of links arriving at a node of degree j, so that

$$q_j = \frac{j\, n\, P(j)}{\sum_{i=1}^{\infty} i\, n\, P(i)} = \frac{j\, P(j)}{z}. \tag{4.2}$$

The probability for this node to have a degree $j \leq d$ is therefore

$$q_d^{'} = \frac{\sum_{j=1}^{d} jP(j)}{\sum_{j=1}^{\infty} jP(j)}. \tag{4.3}$$

By definition, a node with degree d is a degree leader if all of its d neighbors have a degree smaller or equal to d. By using the statistical independence of the degrees of these d neighbors, p_d is found by multiplying Eq. (4.3) d times. ∎

In general, p_d is a function of d whose behavior may be evaluated numerically by inserting the degree distribution $P(d)$ of the network in Eq.(4.1) and by performing the summations. In the following, however, we would like to derive general properties of p_d that do not depend on the details of $P(d)$.

Power-law distribution. Let us focus on the asymptotic behavior of p_d, when d is large, and assume that $P(d)$ may be approximated, for large enough values of d, by a power-law $P(d) \approx d^{-\gamma}$. Let us emphasize that such a tail of the degree distribution is very general, as it includes scale-free distributions (γ finite), while exponential distributions are recovered in the limit $\gamma \to \infty$. In the following, we focus on general values of γ, with the only constraint that $\gamma > 2$ so that the average degree is well-defined.

Theorem 2. *Let $\mathcal{G}$ be a random network with degree distribution $P(d)$ tending for large values of d to $Cd^{-\gamma}$ with $C \in \mathbb{R}_{>0}$ and $\gamma > 2$, then we observe the following transition phase*

$$\lim_{d\to\infty} p_d = \begin{cases} 1 & \text{for} \quad \gamma > 3, \\ e^{-C/z} & \text{for} \quad \gamma = 3, \\ 0 & \text{for} \quad 2 < \gamma < 3. \end{cases} \tag{4.4}$$

Proof: Since $\gamma > 2$, we have $\sum_{j=1}^{\infty} jP(j) = z$ is a finite number and p_d given by Eq.(4.1) asymptotically behaves like

$$\left(1 - \frac{C\sum_{j=d+1}^{\infty} j^{-(\gamma-1)}}{z}\right)^d, \tag{4.5}$$

where we used the fact that $\sum_{j=1}^{d} jP(j) = \sum_{j=1}^{\infty} jP(j) - \sum_{j=d+1}^{\infty} jP(j)$.

For large enough values of d, the summation in (4.5) may be replaced by an integral so that p_d asymptotically behaves like

$$\left(1 - C\,\frac{d^{-(\gamma-2)}}{(\gamma-2)z}\right)^d. \tag{4.6}$$

In order to determine the asymptotic behavior of p_d, it is useful to rewrite Eq.(4.6) as

$$e^{d\ln\left(1 - C\,\frac{d^{-(\gamma-2)}}{(\gamma-2)z}\right)}$$

whose dominating term is, when $C\,\frac{d^{-(\gamma-2)}}{(\gamma-2)z}$ is sufficiently small,

$$e^{-C\,\frac{d^{-(\gamma-3)}}{(\gamma-2)z}}.$$

By construction, $\gamma > 2$ and z is positive, so that the asymptotic values of p_d, for large enough values of d, are

$$\lim_{d\to\infty} e^{-C\,\frac{d^{-(\gamma-3)}}{(\gamma-2)z}} = \begin{cases} 1 & \text{for} \quad \gamma > 3, \\ e^{-C/z} & \text{for} \quad \gamma = 3, \\ 0 & \text{for} \quad 2 < \gamma < 3. \end{cases} \qquad \blacksquare$$

The system therefore undergoes a transition at $\gamma = 3$. If the tail of the degree distribution decreases fast enough, so that $\gamma > 3$, the probability p_d asymptotically goes to 1. Consequently, nodes with a higher degree have a larger probability to be a degree leader and, for large enough values of d, any node is a degree leader. When $\gamma < 3$, in contrast, the probability to be a degree leader decreases with the degree d and asymptotically vanishes, so that, surprisingly, nodes with a larger degree might have a smaller probability to be a degree leader. This result, that may appear intriguing at first sight, comes from the fact that a node with a high degree has a large number of neighbors to compare with and a large number of conditions to satisfy in order to be a local leader (see the exponent d in Eq. (4.1)). Consequently, the probability that it connects to a node with a still higher connectivity might be large. In contrast, it is very probable that such a node with a high degree has a larger degree than each of its neighbors, as seen in Eq.(4.3), where $q_d^{'}$ is monotonically increasing function of d. The transition (4.4) from a *rich gets richer* situation to a *rich is poor* situation therefore originates in the competition between these two opposite effects.

Strict degree leader. Before going further, let us discuss the case of strict leaders. In that case, the calculations are the same as before, except for the sums in p_d that do not go until d but until $d-1$. However, this difference is vanishingly small for large enough values of d, so that the transition (4.4) is recovered.

4.3 Computer simulations

Validation. In this section, we will verify the validity of the theoretical predictions (4.1) and, especially, the existence of the regime $p_d \to 0$ when $\gamma < 3$. One should first stress that the results derived in the previous section are valid for uncorrelated networks composed of an infinite number of nodes. However, whatever the specified degree distribution $P(d)$, a typical realization of the network (in a computer simulation or in realistic situation) involves only a finite number of nodes. This also implies that the largest degree d_{max} in the network is a finite number. The degree d_{max} of this global leader might be estimated by using tools from the theory of extreme statistics [25], but the main point here is that the global leader is also a degree leader. Consequently, the probability for a node of degree d_{max} to be a degree leader, when measured in such a system, is $P_{d_{max}} = 1$, in contradiction with the prediction $p_d \to 0$.

Finite-size effect. In order to highlight this finite-size effect with computer simulations, it is helpful to consider the truncated power laws defined by

$$\begin{aligned} P(d) &= \bar{C} d^{-\gamma} \quad \text{for} \quad d \leq d_{max}, \\ P(d) &= 0 \quad \text{otherwise}, \end{aligned} \tag{4.7}$$

where the constant of normalization depends on γ and on the cut-off d_{max}, $\bar{C} = 1/\sum_{i=1}^{d_{max}} d^{-\gamma}$. Such degree distributions offer the possibility to tune the value of the extremal degree d_{max} together with a particularly simple expression for $P(d)$. In order to generate numerically random uncorrelated networks with the specified degree distribution (4.7), we proceed as follows [18]. We assign to each node i in a set of n nodes a degree $\boldsymbol{d}_i$ extracted from the probability distribution (4.7) and impose that $\sum_{i=1}^{n} \boldsymbol{d}_i$ is even. Then, the network is constructed by randomly assigning the $m = \sum_{i=1}^{n} \boldsymbol{d}_i/2$ edges while respecting the pre-assigned degrees $\boldsymbol{d}_i$. In the simulations, we have considered networks with $n = 10^5$ nodes and averaged the results over 100 realizations of the random process. One should also stress that we have only considered truncated distributions such that d_{max} is effectively the maximum degree for each realization of the network, i.e., such that the expected number of nodes with d_{max} verifies $nP(d_{max}) > 1$. Computer simulations (see Fig. 4.2) show an excellent agreement with the theoretical prediction (4.1) and confirm that p_d first decreases to values close to 0, as predicted by (4.4), before increasing to 1 due to finite size effects.

In order to evaluate where finite size effects become non-negligible, we have focused on the value d_c where P_d is minimum and studied the relation between d_c and d_{max}. By inserting the distribution (4.7) and integrating numerically (4.1), one observes that d_c increases linearly with d_{max}, $d_c \approx \alpha d_{max}$. When $\gamma = 2.2$, for instance, one finds $\alpha = 0.3189$. This linear dependence has important consequences as it implies that finite size effects only affect a vanishingly small number of the nodes when d_{max} is sufficiently large. To show this, let us consider the proportion $P(FS)$ of nodes that are affected by the finite size effects

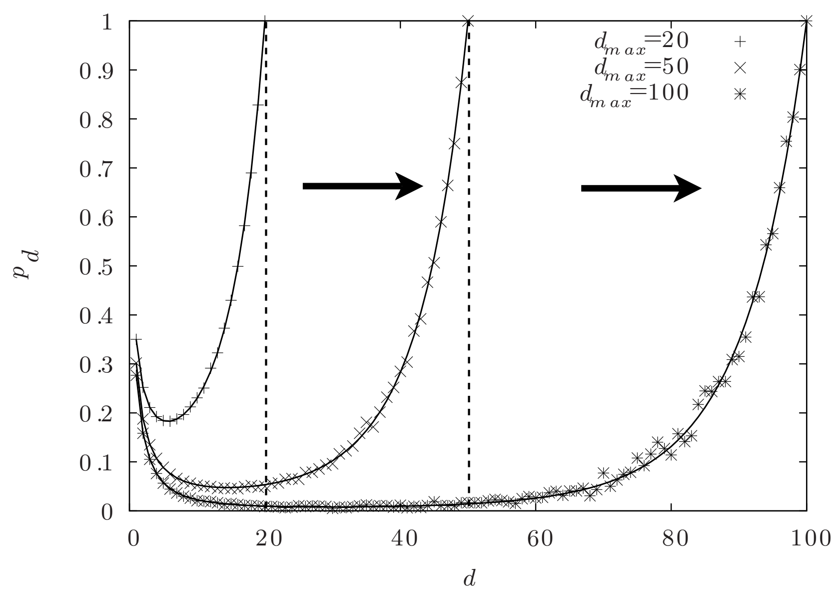


Figure 4.2: The probability p_d measured in random networks composed of 10^5 nodes and whose degree distribution is a truncated power law (4.7) with $\gamma = 2.2$. The results are averaged 100 times. The solid lines are the theoretical prediction (4.1), evaluated numerically for the degree distributions (4.7). The value of d where p_d begins to increase toward $p_d = 1$ due to finite-size effects (see main text) is seen to be proportional with d_{max}.

$$\begin{aligned} P(FS) &= \sum_{i=\alpha d_{max}}^{d_{max}} \bar{C} i^{-\gamma} \approx \int_{i=\alpha d_{max}}^{d_{max}} \bar{C} i^{-\gamma} \\ &= \frac{\bar{C}}{\gamma - 1} (\alpha^{-(\gamma-1)} - 1) d_{max}^{-(\gamma-1)}, \end{aligned} \tag{4.8}$$

where the summation has been replaced by an integral, as d_{max} is sufficiently large. The quantity $P(FS)$ obviously goes to zero when $d_{max} \to \infty$.

Behaviour near the maximal degree. Before concluding, let us also derive the behavior of p_d close to d_{max}. In that case, numerical integration shows an exponential decrease in $(d_{max} - d)$ so that one expects a solution of the form

$$P_d \approx e^{D(d_{max}-d)}, \tag{4.9}$$

for some constant $D< 0$. Let us derive Eq.(4.9) from

$$p_d = e^{d \ln\left(1 - \bar{C}(\sum_{j=d+1}^{d_{max}} j^{-(\gamma-1)})/z\right)}. \tag{4.10}$$

By looking at the dominant terms for small values of $d^{'} := d_{max} - d$ and d_{max} sufficiently large, we obtain

$$\begin{aligned} p_d &\approx e^{-\bar{C} d_{max} (\sum_{j=d+1}^{d_{max}} j^{-(\gamma-1)})/z}, \\ &\approx e^{-\bar{C} d_{max} (d_{max}-d) d_{max}^{-(\gamma-1)}/z}, \\ &\approx e^{-\bar{C} \frac{d_{max}^{-(\gamma-2)}}{z} (d_{max}-d)}, \end{aligned}$$

and the constant D in Eq. (4.9) is given by $-\bar{C} d_{max}^{-(\gamma-2)}/z$. This asymptotic behavior has been successfully compared with computer simulations.

4.4 Upper bounds

In this section, we are interested in upper bounds on the probability p that a randomly chosen node in a random network is a strict degree leader ($p = \sum_{d \in \mathbb{N}} p_d {\cdot} P(d)$). These bounds will depend on the minimum degree d_{min}, the maximal degree d_{max} and on the degree distribution $P(d)$.

Lower bounds. Nontrivial lower bounds are difficult to determine since the number of Leaders can be very low. For example, a clique has no strict degree leaders, a star has one degree leaders and a lot of other structures may only contain zero or one leader. Two phenomena can cause degree leaders to vanish: first a super hub dominating all nodes and second a hierarchical structure where you can always find someone stronger than you. Therefore, the proportion of degree leaders can be close to zero.

Strict degree leaders. In the case of upper bounds, the probability to choose a degree leader (not strict) is one for many regular networks (cliques, rings, etc.). Therefore, we restrict ourselves to the case of strict degree leaders.

Let us partition the set of nodes $\mathcal{N}$ into the strict degree leaders and the others that we call the followers, we respectively have the sets $\mathcal{S}$ and $\mathcal{F}$. Moreover, let $\mathcal{N}(i)$ and $\mathcal{N}_{\mathcal{S}}(i)$ be respectively the set of neighbors of the node i and the same set for neighbors who are strict degree leaders. The next results are derived from the following lemma that gives an exact formula for p in function of the strict degree leaders' degrees $\boldsymbol{d}_i$.

Lemma 1. *The probability p for a node to be a strict degree leader in a network is given by*

$$p = \frac{1}{n} \sum_{j \in \mathcal{F}} \sum_{i \in \mathcal{N}_{\mathcal{S}}(j)} \frac{1}{\boldsymbol{d}_i}. \tag{4.11}$$

Proof: Since two strict degree leaders cannot be neighbors, the summation $\sum_{j \in \mathcal{F}} \sum_{i \in \mathcal{N}_{\mathcal{S}}(j)}$ is made over every link (i,j) of the network that connects one strict degree leader i and one follower j. It is therefore equivalent to the summation $\sum_{i \in \mathcal{S}} \sum_{j \in \mathcal{N}(i)}$ and we have:

$$\begin{aligned} \frac{1}{n} \sum_{j \in \mathcal{F}} \sum_{i \in \mathcal{N}_{\mathcal{S}}(j)} \frac{1}{\boldsymbol{d}_i} &= \frac{1}{n} \sum_{i \in \mathcal{S}} \sum_{j \in \mathcal{N}(i)} \frac{1}{\boldsymbol{d}_i} \\ &= \frac{1}{n} \sum_{i \in \mathcal{S}} \frac{1}{\boldsymbol{d}_i} \sum_{j \in \mathcal{N}(i)} 1 \\ &= \frac{1}{n} \sum_{i \in \mathcal{S}} \frac{1}{\boldsymbol{d}_i} \cdot \boldsymbol{d}_i = \frac{|\mathcal{S}|}{n} = p. \end{aligned}$$

■

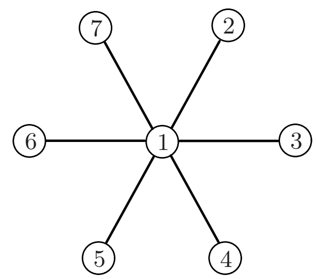


Figure 4.3: That star has one strict degree leader in node 1.

Fig. 4.3 illustrates Lemma 1 where we have

$$\begin{aligned} p &= \frac{1}{n}\sum_{j\in\mathcal{F}}\sum_{i\in\mathcal{N}_{\mathcal{S}}(j)}\frac{1}{\boldsymbol{d}_i} \\ &= \frac{1}{7}\sum_{j\in\mathcal{F}}\frac{1}{6} \\ &= \frac{1}{7}\frac{1}{6}\cdot|\{j\in\mathcal{F}\}| = \frac{1}{7}\frac{1}{6}\,6 = \frac{1}{7}. \end{aligned}$$

Proposition 3. *The probability* p *to be a strict degree leader in a random network with degree distribution* $P(d)$*,* n *nodes and* m *links, have the following upper bounds*

$$p \leq \frac{m/n}{d_{min}+1}, \tag{4.12}$$

$$p \leq \frac{d_{max}-1}{2d_{max}-1}, \tag{4.13}$$

$$p \leq \frac{d_{min}+1}{2d_{min}+1}\sum_{d=d_{min}}^{d_{max}-1} P(d)\,\frac{d}{d+1}. \tag{4.14}$$

Proof: The first upper bound is established from Lemma 1 and the fact that the lowest degree of a strict degree leader is $d_{min}+1$:

$$\begin{aligned} p &= \frac{1}{n}\sum_{j\in\mathcal{F}}\sum_{i\in\mathcal{N}_{\mathcal{S}}(j)}\frac{1}{\boldsymbol{d}_i} \\ &\leq \frac{1}{n}\cdot\frac{1}{d_{min}+1}\sum_{j\in\mathcal{F}}\sum_{i\in\mathcal{N}_{\mathcal{S}}(j)}1 \\ &\leq \frac{1}{n}\cdot\frac{1}{d_{min}+1}\cdot m. \end{aligned}$$

Using again Lemma 1 and the fact that a link (i,j) from a leader to a follower implies $\boldsymbol{d}_i \geq \boldsymbol{d}_j+1$, we have

$$\begin{aligned} p &= \frac{1}{n}\sum_{j\in\mathcal{F}}\sum_{i\in\mathcal{N}_{\mathcal{S}}(j)}\frac{1}{\boldsymbol{d}_i} \\ &\leq \frac{1}{n}\sum_{j\in\mathcal{F}}\sum_{i\in\mathcal{N}_{\mathcal{S}}(j)}\frac{1}{\boldsymbol{d}_j+1} \\ &\leq \frac{1}{n}\sum_{j\in\mathcal{F}}\frac{\boldsymbol{d}_j}{\boldsymbol{d}_j+1} \\ &\leq \frac{1}{n}\sum_{d\in\mathbb{N}}|\{j\in\mathcal{F}:\boldsymbol{d}_j=d\}|\cdot\frac{d}{d+1} \\ &= \sum_{d=d_{min}}^{d_{max}-1}P(d)(1-p_d)\cdot\frac{d}{d+1}, \end{aligned} \tag{4.15}$$

where $|\{j\in\mathcal{F}:\boldsymbol{d}_j=d\}|$ corresponds to the number of followers with degree d and $P(d)(1-p_d)$ is the proportion of followers with degree d. We also use the fact that the degrees of the followers range from d_{min} to $d_{max}-1$. Hence the two upper bounds in Eq. (4.13-4.14) can be deduced. By normalizing Ineq. (4.15) by the sum $\sum_{d\in\mathbb{N}}P(d)(1-p_d)$ that is equal to $1-p$, we have

$$\begin{aligned} p &\leq (1-p)\sum_{d=d_{min}}^{d_{max}-1}\frac{P(d)(1-p_d)}{(1-p)}\cdot\frac{d}{d+1} \\ &\leq (1-p)\cdot\frac{d_{max}-1}{d_{max}} \\ &\leq \frac{d_{max}-1}{2d_{max}-1}. \end{aligned}$$

By knowing that a strict degree leader has a degree at least equals to

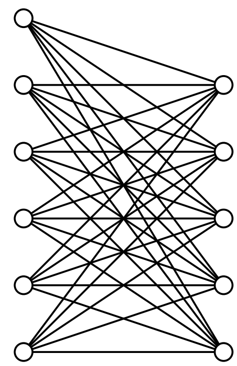


Figure 4.4: This bipartite network $K_{6,5}$ has $d_{max} = 6$, $d_{min} = 5$, $n = 11$ and $m = 30$. Its proportion of strict degree leaders is 5/11.

$d_{min} + 1$, we obtain for the last upper bound

$$\begin{aligned} p &\leq \sum_{d=d_{min}}^{d_{max}-1} P(d)\frac{d}{d+1} - \sum_{d=d_{min}}^{d_{max}-1} P(d)p_d\frac{d}{d+1} \\ &\leq \sum_{d=d_{min}}^{d_{max}-1} P(d)\frac{d}{d+1} - \frac{d_{min}}{d_{min}+1}p \\ &\leq \frac{d_{min}+1}{2d_{min}+1}\sum_{d=d_{min}}^{d_{max}-1} P(d)\frac{d}{d+1}. \end{aligned}$$

■

The three upper bounds given in Proposition 3 can be achieved as shown in the example of Figure 4.4. In particular, the probability p is always upper bounded by 1/2 and tends to that value for bipartite networks $K_{n+1,n}$ when n tends to infinity.

4.5 Conclusions

Results. In this chapter, we have focused on the statistical properties of degree leaders. Such nodes, that may be viewed as local hubs, have a crucial location in a social or information network, as they dominate all of their neighbors. Their identification and a better understanding of their properties might therefore be of practical interest. In marketing, for instance, degree leaders are good candidates to target in order to maximize a marketing campaign or to minimize the erosion of customers from a company, e.g., *churn* for mobile operators [4]. We have observed that the probability for a node of degree d to be a degree leader undergoes a transition from a *rich gets richer* to a *rich is poor* situation, that suggests that nodes with a high degree might not be the most influential at the local level. It is interesting to stress that the transition takes place at a realistic value of the power-law exponent $\gamma = 3$ [54, 56], i.e., scale-free distributions usually have an exponent between 2 and 3 [78], and that $\gamma = 3$ is also the critical value under which fluctuations $< (d-z)^2 >$ around the average degree z diverge.

Future research. To conclude, one should stress that the local maxima of other node quantities could also give insight into the network structure, e.g., the number of triangles [13]. More general definitions of degree leaders could also be considered, e.g., a node of degree d is a α-leader if all of its neighbors have a degree $d^{'} < d/\alpha$. A generalization of our study to such situations and a comparison with empirical data (where nodes might exhibit degree-degree correlations) could therefore be of interest.

Chapter 5

Maximization of PageRank via Outlinks

In this chapter, we analyze **linkage strategies** for a set $\mathcal{I}$ of webpages for which the webmaster wants to maximize the sum of Google's PageRank scores. The webmaster can only choose the hyperlinks *starting* from the webpages of $\mathcal{I}$ and has no control on the hyperlinks from other webpages. We provide an optimal linkage strategy under some reasonable assumptions.

The optimal linkage strategy is deduced from the optimal outlink structure and the optimal inlink structure for the set of webpages we consider. Moreover some **sensitivity results** on the PageRank when one adds or removes a link are given, and the results are extended to the case where we consider a larger family of stochastic matrices (where the links have different weights) including the Google matrix as a particular case.

5.1 Introduction

Motivation. PageRank, a measure of webpages' relevance introduced by Brin and Page, is at the heart of the well known search engine Google [15, 85]. Google classifies the webpages according to the pertinence scores given by PageRank, which are computed from the network structure of the Web. A page with a high PageRank will appear among the first items in the list of pages corresponding to a particular query.

If we look at the popularity of Google, it is not surprising that some webmasters want to increase the PageRank of their webpages in order to get more visits from websurfers to their website. Since PageRank is based on the link structure of the Web, it is therefore useful to understand how addition or deletion of hyperlinks influence it.

Mathematical analysis of PageRank's sensitivity with respect to perturbations of the matrix describing the webnetwork is a typical subject of interest (see for instance [6, 10, 50, 58, 60, 64] and the references therein). Normwise and componentwise conditioning bounds [50] as well as the derivative [58, 60] are used to understand the sensitivity of the PageRank vector. It appears that the PageRank vector is relatively insensitive to small changes in the network structure, at least when these changes concern webpages with a low PageRank score [10, 58]. One could think therefore that trying to modify one's PageRank via changes in the link structure of the Web is a waste of time. However, what is important for webmasters is not the values of the PageRank vector but the *ranking* that ensues from it. Indeed, the relevance of webpages are compared in a list ordered by their PageRanks, that is the ordinal ranking. Lempel and Morel [64] showed that PageRank is not rank-stable, i.e., small modifications in the link structure of the webnetwork may cause dramatic changes in the ordinal ranking of the webpages. On the other hand, the computation part of the problem has been analyzed by Ipsen and Wills who propose efficient criteria to guarantee correct ordinal ranking [39]. Therefore, the question of how the PageRank of a particular page or set of pages could be increased–even slightly–by adding or removing links to the webnetwork remains of interest. The same question for the modified PageRank (where links have different weights) like for instance in [2] deserves the same attention.

Related results. As it is well known [5, 38], if a hyperlink from a page i to a page j is added, with no other modification in the Web, then the PageRank of j will increase. But in general, you do not have control on the *inlinks* of your webpage unless you pay another webmaster to add a hyperlink from his/her page to your or you make an *alliance* with him/her by trading a link for a link [7, 30]. But it is natural to ask how you could modify your PageRank by yourself. This leads to analyze how the choice of the *outlinks* of a page can influence its own PageRank. Sydow [97] showed via numerical simulations that adding well chosen outlinks to a webpage may increase significantly one's PageRank ranking. Avrachenkov and Litvak [6] analyzed theoretically the possible effect of new outlinks on the PageRank of a page and its neighbors. Supposing that a webpage has control only on its outlinks, they gave the optimal linkage strategy for this single page. Bianchini et al. [10] as well as Avrachenkov and Litvak in [5] consider the impact of links between web communities (websites or sets of related webpages), respectively on the sum of the PageRanks and on the individual PageRank scores of the pages of some community. They give general rules in order to have a PageRank as high as possible but they do not provide an optimal link structure for a website.

Goals of the study. Our aim in this chapter is to find a generalization of Avrachenkov–Litvak's optimal linkage strategy [6] to the case of *a website with several pages*. We consider a given set of pages and suppose we have only control on the *outlinks* of these pages. We are interested in the problem of *maximizing the sum of the PageRanks* of these pages.

Suppose $\mathcal{G} = (\mathcal{N}, \mathcal{L})$ is the webnetwork, with a set of nodes $\mathcal{N} = \{1, \ldots, n\}$ and a set of links $\mathcal{L} \subseteq \mathcal{N} \times \mathcal{N}$. For a subset of nodes $\mathcal{I} \subseteq \mathcal{N}$, we define

$$\begin{aligned}
\mathcal{L}_{\mathcal{I}} &= \{(i,j) \in \mathcal{L} \colon i,j \in \mathcal{I}\} \text{ the set of internal links,} \\
\mathcal{L}_{\mathrm{out}(\mathcal{I})} &= \{(i,j) \in \mathcal{L} \colon i \in \mathcal{I}, j \notin \mathcal{I}\} \text{ the set of external outlinks,} \\
\mathcal{L}_{\mathrm{in}(\mathcal{I})} &= \{(i,j) \in \mathcal{L} \colon i \notin \mathcal{I}, j \in \mathcal{I}\} \text{ the set of external inlinks,} \\
\mathcal{L}_{\overline{\mathcal{I}}} &= \{(i,j) \in \mathcal{L} \colon i,j \notin \mathcal{I}\} \text{ the set of external links.}
\end{aligned}$$

For example, the network given in Fig. 5.3 has 3 internal links, $\mathcal{L}_{\mathcal{I}} = \{(1,3),(1,2),(2,3)\}$, one external outlink, $\mathcal{L}_{\mathrm{out}(\mathcal{I})} = \{(3,4)\}$, one external inlink, $\mathcal{L}_{\mathrm{in}(\mathcal{I})} = \{(7,1)\}$ and 12 external links, $\mathcal{L}_{\overline{\mathcal{I}}} = \{(i,j) : 4 \leq i,j \leq 7,\ i \neq j\}$.

If we do not impose any condition on $\mathcal{L}_{\mathcal{I}}$ and $\mathcal{L}_{\mathrm{out}(\mathcal{I})}$, the problem of maximizing the sum of the PageRanks of pages of $\mathcal{I}$ is quite trivial and does not have much interest (see the discussion in Section 5.3). Therefore, when characterizing optimal link structures, we will make the following *accessibility assumption*: every page of the website must have access to the rest of the Web.

Our first main result concerns the *optimal outlink structure* for a given website. In the case where the subnetwork corresponding to the website is strongly connected, Theorem 4 can be specialized as follows.

Theorem. *Let $\mathcal{L}_{\mathcal{I}}$, $\mathcal{L}_{\text{in}(\mathcal{I})}$ and $\mathcal{L}_{\overline{\mathcal{I}}}$ be given. Suppose that the subnetwork $(\mathcal{I}, \mathcal{L}_{\mathcal{I}})$ is strongly connected and $\mathcal{L}_{\mathcal{I}} \neq \emptyset$. Then every optimal outlink structure $\mathcal{L}_{\text{out}(\mathcal{I})}$ is to have only one outlink to a particular page outside of $\mathcal{I}$.*

We are also interested in the optimal *internal* link structure for a website. In the case where there is a unique leaking node in the website, that is only one node linking to the rest of the web, Theorem 5 can be specialized as follows.

Theorem. *Let $\mathcal{L}_{\text{out}(\mathcal{I})}$, $\mathcal{L}_{\text{in}(\mathcal{I})}$ and $\mathcal{L}_{\overline{\mathcal{I}}}$ be given. Suppose that there is only one leaking node in $\mathcal{I}$. Then every optimal internal link structure $\mathcal{L}_{\mathcal{I}}$ is composed of a forward chain of links together with every possible backward link.*

Putting together the two theorems above, we get in Theorem 6 the *optimal link structure* for a website. This optimal structure is illustrated in Fig. 5.1.

Theorem. *Let $\mathcal{L}_{\text{in}(\mathcal{I})}$ and $\mathcal{L}_{\overline{\mathcal{I}}}$ be given. Then, for every optimal link structure, $\mathcal{L}_{\mathcal{I}}$ is composed of a forward chain of links together with every possible backward link, and $\mathcal{L}_{\text{out}(\mathcal{I})}$ consists of a unique outlink, starting from the last node of the chain.*

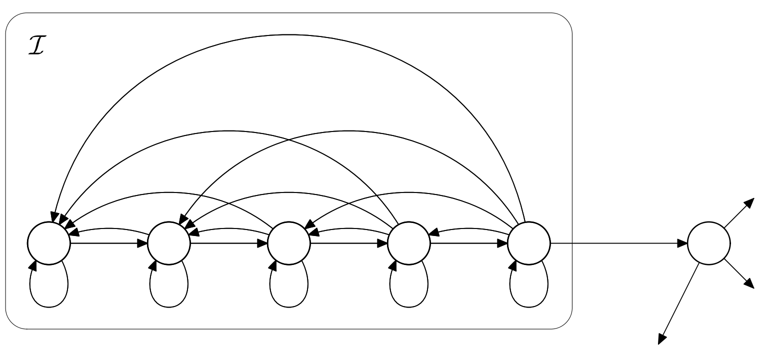


Figure 5.1: Every optimal linkage strategy for a set $\mathcal{I}$ of five pages must have this structure.

Structure. This chapter is organized as follows. In the following preliminary section, we recall some network concepts as well as the definition of the PageRank, and we introduce some notations. In Section 5.2, we develop tools for analyzing the PageRank of a set of pages $\mathcal{I}$. Then we come to the main part of this chapter: in Section 5.3 we provide the optimal linkage strategy for a set of nodes. In Section 5.4, we give some extensions and variants of the main theorems. We end this chapter with some concluding remarks.

5.2 PageRank of a website

We have seen in Section 2.3 that the PageRank vector $\boldsymbol{\pi}$ is defined by

$$\begin{aligned}\boldsymbol{\pi}^T &= \boldsymbol{\pi}^T G,\\ \boldsymbol{\pi}^T \mathbf{1} &= 1,\end{aligned}$$

where $G = cS + (1-c)\mathbf{1}\boldsymbol{z}^T$ is the Google matrix and $S = [S_{ij}]_{i,j\in\mathcal{N}}$ is the scaled adjacency matrix of the network (see Eq. (2.3)). Remember that it is assumed that *each node has at least one outlink*, i.e., the outdegree $\boldsymbol{d}_i \neq 0$ for every $i \in \mathcal{N}$.

We will also remark that some of the first results in this chapter remain valid for the more general case $G = cP + (1-c)\mathbf{1}\boldsymbol{z}^T$ with P representing any row stochastic matrix, i.e., P is nonnegative with $P\mathbf{1} = \mathbf{1}$. As said in the introduction, this allows us to personalize the weights of the links in the networks so that the outlinks of node i equally weighted by $1/\boldsymbol{d}_i$ in the matrix S are now weighted by P_{ij}.

We are interested in characterizing the *PageRank of a set* $\mathcal{I}$. We define this as the sum

$$\boldsymbol{\pi}^T \boldsymbol{e}_{\mathcal{I}} = \sum_{i\in\mathcal{I}} \boldsymbol{\pi}_i,$$

where $\boldsymbol{e}_{\mathcal{I}}$ denotes the vector with a 1 in the entries of $\mathcal{I}$ and 0 elsewhere. Note that the PageRank of a set corresponds to the notion of energy of a community in [10].

Let $\mathcal{I} \subseteq \mathcal{N}$ be a subset of the nodes of the network. The PageRank of $\mathcal{I}$ can be expressed as $\boldsymbol{\pi}^T \boldsymbol{e}_{\mathcal{I}} = (1-c)\boldsymbol{z}^T(I-cS)^{-1}\boldsymbol{e}_{\mathcal{I}}$ from PageRank Eq. (2.5). Let us then define the vector

$$\boldsymbol{v} = (I - cS)^{-1}\boldsymbol{e}_{\mathcal{I}}. \tag{5.1}$$

With this, we have the following expression for the PageRank of the set $\mathcal{I}$:

$$\boldsymbol{\pi}^T \boldsymbol{e}_{\mathcal{I}} = (1-c)\boldsymbol{z}^T \boldsymbol{v}. \tag{5.2}$$

The vector $\boldsymbol{v}$ will play a crucial role throughout this chapter. In this section, we will first present a probabilistic interpretation for this vector and prove some of its properties. We will then show how it can be used in order to analyze the influence of some page $i \in \mathcal{I}$ on the PageRank of the set $\mathcal{I}$. We will end this section by briefly introducing the concept of basic absorbing network, which will be useful in order to analyze optimal linkage strategies under some assumptions.

Mean number of visits before zapping. Let us first see how the entries of the vector $\boldsymbol{v} = (I - cS)^{-1}\boldsymbol{e}_{\mathcal{I}}$ can be interpreted. Let us consider a random surfer on the webnetwork $\mathcal{G}$ that, as described in Section 2.4, follows the hyperlinks of the webnetwork with a probability c. But, instead of zapping to some page of $\mathcal{G}$ with probability $(1 - c)$, he *stops* his walk with probability $(1 - c)$ at each time step. This is equivalent to considering a random walk on the extended network $\mathcal{G}_e = (\mathcal{N} \cup \{n+1\}, \mathcal{L} \cup \{(i, n+1) \colon i \in \mathcal{N}\})$ with a transition probability matrix

$$S_e = \begin{pmatrix} cS & (1-c)\mathbf{1} \\ 0 & 1 \end{pmatrix}.$$

At each time step, with probability $1-c$, the random surfer can *disappear* from the original network, that is he can reach the absorbing node $n+1$.

The nonnegative matrix $(I - cS)^{-1}$ is commonly called the fundamental matrix of the absorbing Markov chain defined by S_e (see for instance [43, 92]). In the extended network $\mathcal{G}_e$, the entry $[(I - cS)^{-1}]_{ij}$ is the expected number of visits to node j before reaching the absorbing node $n + 1$ when starting from node i. From the point of view of the random surfer described in Section 2.4, the entry $[(I - cS)^{-1}]_{ij}$ is the expected number of visits to node j before zapping for the first time when starting from node i.

Therefore, the vector $\boldsymbol{v}$ defined in equation (5.1) has the following probabilistic interpretation. The entry $\boldsymbol{v}_i$ is the *expected number of visits to the set $\mathcal{I}$ before zapping* for the first time when the random surfer starts his walk in node i.

Let us remark that all previous comments about the vector $\boldsymbol{v}$ remain valid if we replace the matrix S by any row stochastic matrix P. The three next Lemmas presented in the next section also hold for that more general case as we will see.

Three lemmas. Let us first prove some simple properties about the vector $\boldsymbol{v}$.

Lemma 2. *Let* $\boldsymbol{v} \in \mathbb{R}^n_{\geq 0}$ *be defined by* $\boldsymbol{v} = cS\boldsymbol{v} + \boldsymbol{e}_{\mathcal{I}}$*. Then,*

(a) $\max_{i \notin \mathcal{I}} \boldsymbol{v}_i \leq c \max_{i \in \mathcal{I}} \boldsymbol{v}_i$,

(b) $\boldsymbol{v}_i \leq 1 + c\,\boldsymbol{v}_i$ *for all* $i \in \mathcal{N}$*; with equality if and only if* $i \in \mathcal{I}$ *and node* i *does not have access to* $\overline{\mathcal{I}}$,

(c) $\boldsymbol{v}_i \geq \min_{j \leftarrow i} \boldsymbol{v}_j$ *for all* $i \in \mathcal{I}$*; with equality if and only if the node* i *does not have access to* $\overline{\mathcal{I}}$*;*

Proof: (a) Since $c < 1$, for all $i \notin \mathcal{I}$,

$$\max_{i \notin \mathcal{I}} \boldsymbol{v}_i = \max_{i \notin \mathcal{I}} \left(c \sum_{j \leftarrow i} \frac{\boldsymbol{v}_j}{\boldsymbol{d}_i} \right) \leq c \max_j \boldsymbol{v}_j .$$

Since $c < 1$, it then follows that $\max_j \boldsymbol{v}_j = \max_{i \in \mathcal{I}} \boldsymbol{v}_i$.

(b) The inequality $\boldsymbol{v}_i \leq \frac{1}{1-c}$ follows directly from

$$\max_i \boldsymbol{v}_i \leq \max_i \left(1 + c \sum_{j \leftarrow i} \frac{\boldsymbol{v}_j}{\boldsymbol{d}_i} \right) \leq 1 + c \max_j \boldsymbol{v}_j .$$

Then the equality $\boldsymbol{v}_i = \frac{1}{1-c}$ occurs if and only if $\boldsymbol{v}_j = \frac{1}{1-c}$ for every $j \leftarrow i$. Indeed, that comes from

$$1 + c\,\boldsymbol{v}_i = \boldsymbol{v}_i = 1 + c \sum_{j \leftarrow i} \frac{\boldsymbol{v}_j}{\boldsymbol{d}_i} .$$

Moreover from (a) it follows that $\boldsymbol{v}_k \leq \frac{c}{1-c}$ for all $k \notin \mathcal{I}$. We necessarily have that i and its children must be in $\mathcal{I}$. By induction, every node k such that i has access to k must belong to $\mathcal{I}$.

(c) Let $i \in \mathcal{I}$. Then, by (b)

$$1 + c\,\boldsymbol{v}_i \geq \boldsymbol{v}_i = 1 + c \sum_{j \leftarrow i} \frac{\boldsymbol{v}_j}{\boldsymbol{d}_i} \geq 1 + c \min_{j \leftarrow i} \boldsymbol{v}_j ,$$

so $\boldsymbol{v}_i \geq \min_{j \leftarrow i} \boldsymbol{v}_j$ for all $i \in \mathcal{I}$. If $\boldsymbol{v}_i = \min_{j \leftarrow i} \boldsymbol{v}_j$ then also $1 + c\,\boldsymbol{v}_i = \boldsymbol{v}_i$ and hence, by (b), the node i does not have access to $\overline{\mathcal{I}}$. ■

Lemma 2 holds when S is replaced by P. The proof is verbatim the same if you change

$$\sum_{j \leftarrow i} \frac{\boldsymbol{v}_j}{\boldsymbol{d}_i} \qquad \text{to} \qquad \sum_{j \leftarrow i} P_{ij}\,\boldsymbol{v}_j , \tag{5.3}$$

where we consider now a weighted average instead of an average.

Let us denote the set of nodes of $\overline{\mathcal{I}}$ which on average give the most visits to $\mathcal{I}$ before zapping by

$$\mathcal{V} = \operatorname*{argmax}_{j \in \overline{\mathcal{I}}} \boldsymbol{v}_j.$$

Then the following lemma is quite intuitive. It says that, among the nodes of $\overline{\mathcal{I}}$, those that provide the higher mean number of visits to $\mathcal{I}$ are parents of $\mathcal{I}$, i.e., parents of some node of $\mathcal{I}$.

Lemma 3 (Parents of $\mathcal{I}$)**.** *If $\mathcal{L}_{\mathrm{in}(\mathcal{I})} \neq \emptyset$, then*

$$\mathcal{V} \subseteq \{j \in \overline{\mathcal{I}} \colon \text{ there exists } \ell \in \mathcal{I} \text{ such that } (j, \ell) \in \mathcal{L}_{\mathrm{in}(\mathcal{I})}\}.$$

If $\mathcal{L}_{\mathrm{in}(\mathcal{I})} = \emptyset$, then $\boldsymbol{v}_j = 0$ for every $j \in \overline{\mathcal{I}}$.

Proof: Suppose first that $\mathcal{L}_{\mathrm{in}(\mathcal{I})} \neq \emptyset$. Let $k \in \mathcal{V}$ and $\boldsymbol{v} = (I - cS)^{-1}\boldsymbol{e}_{\mathcal{I}}$. If we supposed that there does not exist $\ell \in \mathcal{I}$ such that $(k, \ell) \in \mathcal{L}_{\mathrm{in}(\mathcal{I})}$, then we would have, since $\boldsymbol{v}_k > 0$,

$$\boldsymbol{v}_k = c \sum_{j \leftarrow k} \frac{\boldsymbol{v}_j}{\boldsymbol{d}_k} \leq c \max_{j \notin \mathcal{I}} \boldsymbol{v}_j = c\boldsymbol{v}_k < \boldsymbol{v}_k,$$

which is a contradiction. Now, if $\mathcal{L}_{\mathrm{in}(\mathcal{I})} = \emptyset$, then there is no access to $\mathcal{I}$ from $\overline{\mathcal{I}}$, so clearly $\boldsymbol{v}_j = 0$ for every $j \in \overline{\mathcal{I}}$. ∎

In the same manner as for Lemma 2, we can use the replacement in (5.3) to prove Lemma 3 for any row stochastic matrix P instead of the particular matrix S.

Lemma 3 shows that the nodes $j \in \overline{\mathcal{I}}$ which provide the higher value of $\boldsymbol{v}_j$ must belong to the set of parents of $\mathcal{I}$. The converse is not true, as we will see in the following example: some parents of $\mathcal{I}$ can provide a lower mean number of visits to $\mathcal{I}$ than other nodes which are not parents of $\mathcal{I}$. In other words, Lemma 3 gives a necessary but not sufficient condition in order to maximize the entry $\boldsymbol{v}_j$ for some $j \in \overline{\mathcal{I}}$.

Example 1. Let us see on an example that having $(j, i) \in \mathcal{L}_{\mathrm{in}(\mathcal{I})}$ for some $i \in \mathcal{I}$ is not sufficient to have $j \in \mathcal{V}$. Consider the network in Fig. 5.2. Let $\mathcal{I} = \{1\}$ and take a damping factor $c = 0.85$. For $\boldsymbol{v} = (I - cS)^{-1}\boldsymbol{e}_1$, we have

$$\boldsymbol{v}_2 = \boldsymbol{v}_3 = \boldsymbol{v}_4 = 4.359 > \boldsymbol{v}_5 = 3.521 > \boldsymbol{v}_6 = 3.492 > \boldsymbol{v}_7 > \cdots > \boldsymbol{v}_{11},$$

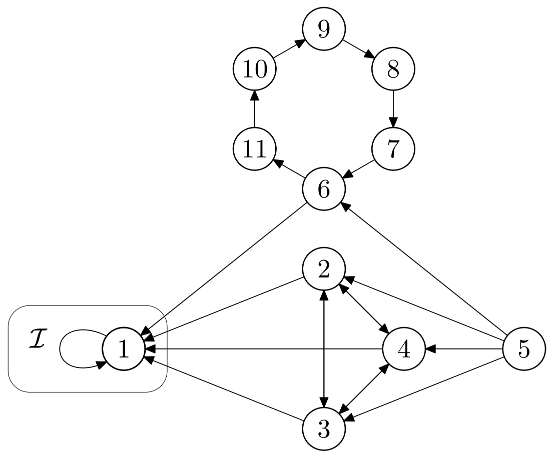


Figure 5.2: The node $6 \notin \mathcal{V}$ and yet it is a parent of $\mathcal{I} = \{1\}$ (see Example 1).

so $\mathcal{V} = \{2, 3, 4\}$. As ensured by Lemma 3, every node of the set $\mathcal{V}$ is a parent of node 1. But here, $\mathcal{V}$ does not contain all parents of node 1. Indeed, the node $6 \notin \mathcal{V}$ while it is a parent of 1 and is moreover its parent with the lowest outdegree. Moreover, we see in this example that node 5, which is a not a parent of node 1 but a parent of node 6, gives a higher value of the expected number of visits to $\mathcal{I}$ before zapping, than node 6, parent of 1. Let us try to get some intuition about that. When starting from node 6, a random surfer has probability one half to reach node 1 in only one step. But he has also a probability one half to move to node 11 and to be send far away from node 1. On the other hand, when starting from node 5, the random surfer can not reach node 1 in only one step. But with probability 3/4 he will reach one of the nodes 2, 3 or 4 in one step. And from these nodes, the websurfer stays very near to node 1 and can not be sent far away from it.

In the next lemma, we show that from some node $i \in \mathcal{I}$ which has access to $\overline{\mathcal{I}}$, there always exists what we call a *decreasing path* to $\overline{\mathcal{I}}$. That is, we can find a path such that the mean number of visits to $\mathcal{I}$ is higher when starting from some node of the path than when starting from the successor of this node in the path.

Lemma 4 (Decreasing paths to $\overline{\mathcal{I}}$)**.** *For every $i_0 \in \mathcal{I}$ which has access* *to $\overline{\mathcal{I}}$, there exists a path $(i_0, i_1, \ldots, i_s)$ with $i_1, \ldots, i_{s-1} \in \mathcal{I}$ and $i_s \in \overline{\mathcal{I}}$ such that*

$$\boldsymbol{v}_{i_0} > \boldsymbol{v}_{i_1} > \ldots > \boldsymbol{v}_{i_s}.$$

Proof: Let us simply construct a decreasing path recursively by

$$i_{k+1} \in \underset{j \leftarrow i_k}{\operatorname{argmin}} \, \boldsymbol{v}_j,$$

as long as $i_k \in \mathcal{I}$. If i_k has access to $\overline{\mathcal{I}}$, then $\boldsymbol{v}_{i_{k+1}} < \boldsymbol{v}_{i_k} < \frac{1}{1-c}$ by Lemma 2(b) and (c), so the node i_{k+1} has also access to $\overline{\mathcal{I}}$. By assumption, i_0 has access to $\overline{\mathcal{I}}$. Moreover, the set $\mathcal{I}$ has a finite number of elements, so there must exist an s such that $i_s \in \overline{\mathcal{I}}$. ■

Since Lemma 4 uses Lemma 2, which is valid for a general row stochastic matrix P, Lemma 4 is also valid for such matrices.

Influence of the outlinks of a node. We will now see how a modification of the outlinks of some node $i \in \mathcal{N}$ can change the PageRank of a subset of nodes $\mathcal{I} \subseteq \mathcal{N}$. So we will compare two networks on $\mathcal{N}$ defined by their set of links, $\mathcal{L}$ and $\widetilde{\mathcal{L}}$ respectively.

Every item corresponding to the network defined by the set of links $\widetilde{\mathcal{L}}$ will be written with a tilde symbol. So $\widetilde{S}$ and $\widetilde{P}$ denote its row stochastic adjacency matrices (the one from Google and the general one), $\widetilde{\boldsymbol{\pi}}$ the corresponding PageRank vector, $\widetilde{\boldsymbol{d}}_i = |\{j \colon (i,j) \in \widetilde{\mathcal{L}}\}|$ the outdegree of some node i in this network, $\widetilde{\boldsymbol{v}} = (I - c\widetilde{S})^{-1} \boldsymbol{e}_{\mathcal{I}}$ and $\widetilde{\mathcal{V}} = \operatorname{argmax}_{j \in \overline{\mathcal{I}}} \widetilde{\boldsymbol{v}}_j$. Finally, by $j \mathrel{\widetilde{\leftarrow}} i$ we mean $j \in \{k \colon (i,k) \in \widetilde{\mathcal{L}}\}$.

So, let us consider two networks defined respectively by their set of links $\mathcal{L}$ and $\widetilde{\mathcal{L}}$. Suppose that they differ only in the links starting from some given node i, that is $\{j \colon (k,j) \in \mathcal{L}\} = \{j \colon (k,j) \in \widetilde{\mathcal{L}}\}$ for all $k \neq i$. Then their scaled adjacency matrices S and $\widetilde{S}$ are linked by a rank one change. Let us then define the vector

$$\boldsymbol{\delta} = \sum_{j \mathrel{\widetilde{\leftarrow}} i} \frac{\boldsymbol{e}_j}{\widetilde{\boldsymbol{d}}_i} - \sum_{j \leftarrow i} \frac{\boldsymbol{e}_j}{\boldsymbol{d}_i}, \tag{5.4}$$

which gives the change to apply to the row i of the matrix S in order to get $\widetilde{S}$.

Now let us first express the difference between the PageRank of $\mathcal{I}$ for two configurations differing only in the links starting from some node i. Note that in the following lemma the personalization vector $\boldsymbol{z}$ does not appear explicitly in the expression of $\widetilde{\boldsymbol{\pi}}$.

Theorem 3. *Let two networks be defined respectively by $\mathcal{L}$ and $\widetilde{\mathcal{L}}$ and let $i \in \mathcal{N}$ such that for all $k \neq i$, $\{j\colon (k,j) \in \mathcal{L}\} = \{j\colon (k,j) \in \widetilde{\mathcal{L}}\}$. Then*

$$\widetilde{\boldsymbol{\pi}}^T \boldsymbol{e}_{\mathcal{I}} = \boldsymbol{\pi}^T \boldsymbol{e}_{\mathcal{I}} + c\,\boldsymbol{\pi}_i \,\frac{\boldsymbol{\delta}^T \boldsymbol{v}}{1 - c\,\boldsymbol{\delta}^T (I - cS)^{-1} \boldsymbol{e}_i}, \tag{5.5}$$

with

$$1 - c\,\boldsymbol{\delta}^T (I - cS)^{-1} \boldsymbol{e}_i > 0.$$

Proof: Clearly, the scaled adjacency matrices are linked by $\widetilde{S} = S + \boldsymbol{e}_i\,\boldsymbol{\delta}^T$. Since $c < 1$, the matrix $(I - cS)^{-1}$ exists and the PageRank vectors can be expressed as

$$\begin{aligned}\boldsymbol{\pi}^T &= (1-c)\boldsymbol{z}^T (I - cS)^{-1},\\ \widetilde{\boldsymbol{\pi}}^T &= (1-c)\boldsymbol{z}^T (I - c\,(S + \boldsymbol{e}_i \boldsymbol{\delta}^T))^{-1}.\end{aligned}$$

Applying the Sherman–Morrison formula to $((I - cS) - c\boldsymbol{e}_i \boldsymbol{\delta}^T)^{-1}$, we get

$$\widetilde{\boldsymbol{\pi}}^T = (1-c)\boldsymbol{z}^T (I - cS)^{-1} + (1-c)\boldsymbol{z}^T (I - cS)^{-1} \boldsymbol{e}_i \,\frac{c\boldsymbol{\delta}^T (I - cS)^{-1}}{1 - c\boldsymbol{\delta}^T (I - cS)^{-1} \boldsymbol{e}_i},$$

provided that $1 - c\boldsymbol{\delta}^T (I - cS)^{-1} \boldsymbol{e}_i \neq 0$ and Eq. (5.5) follows immediately. It remains to prove that $1 - c\boldsymbol{\delta}^T (I - cS)^{-1} \boldsymbol{e}_i > 0$. Since $c < 1$, it is enough to show that $\boldsymbol{\delta}^T (I - cS)^{-1} \boldsymbol{e}_i \leq 1$ is always satisfied. Let $\boldsymbol{u} = (I - cS)^{-1} \boldsymbol{e}_i$. Then $\boldsymbol{u} - cS\boldsymbol{u} = \boldsymbol{e}_i$ and, by Lemma 2(a) (applied with $\boldsymbol{v} = \boldsymbol{u}$ and $\mathcal{I} = \{i\}$), $\boldsymbol{u}_j \leq \boldsymbol{u}_i$ for all j. So

$$\boldsymbol{\delta}^T \boldsymbol{u} = \sum_{j \,\widetilde{\leftarrow}\, i} \frac{\boldsymbol{u}_j}{\widetilde{\boldsymbol{d}}_i} - \sum_{j \leftarrow i} \frac{\boldsymbol{u}_j}{\boldsymbol{d}_i} \leq \boldsymbol{u}_i - \sum_{j \leftarrow i} \frac{\boldsymbol{u}_j}{\boldsymbol{d}_i} < \boldsymbol{u}_i - c \sum_{j \leftarrow i} \frac{\boldsymbol{u}_j}{\boldsymbol{d}_i} = 1.$$

■

This theorem remains valid if we consider P and $\widetilde{P}$ instead of S and $\widetilde{S}$. In that case the vector $\boldsymbol{\delta}$ is more generally defined by

$$\boldsymbol{\delta}^T := \boldsymbol{e}_i^T \left(\widetilde{P} - P\right), \tag{5.6}$$

and it gives the change to apply to the row i of the matrix P in order to obtain $\widetilde{P}$. The Sherman-Morrison formula can still be used, and since Lemma 2 holds for P the proof remains the same using the replacement in (5.3) for $\boldsymbol{u}$ and the tilde symbol.

Let us now give an equivalent condition in order to increase the PageRank of $\mathcal{I}$ by changing outlinks of some node i. The PageRank of $\mathcal{I}$ increases essentially when the new set of links favors nodes giving a higher mean number of visits to $\mathcal{I}$ before zapping. The next corollary follows from Theorem 3 and the fact that $c < 1$ and $\boldsymbol{\pi} > 0$, moreover it holds for the modified PageRank with P when we use the general definition of $\boldsymbol{\delta}$ in Eq. (5.6).

Corollary 2 (PageRank and mean number of visits before zapping)**.** *Let two networks be respectively defined by $\mathcal{L}$ and $\widetilde{\mathcal{L}}$ and let $i \in \mathcal{N}$ such that for all $k \neq i$, $\{j\colon (k,j) \in \mathcal{L}\} = \{j\colon (k,j) \in \widetilde{\mathcal{L}}\}$. Then*

$$\widetilde{\boldsymbol{\pi}}^T \boldsymbol{e}_{\mathcal{I}} > \boldsymbol{\pi}^T \boldsymbol{e}_{\mathcal{I}} \quad \text{if and only if} \quad \boldsymbol{\delta}^T \boldsymbol{v} > 0$$

and $\widetilde{\boldsymbol{\pi}}^T \boldsymbol{e}_{\mathcal{I}} = \boldsymbol{\pi}^T \boldsymbol{e}_{\mathcal{I}}$ if and only if $\boldsymbol{\delta}^T \boldsymbol{v} = 0$.

The next two subsections give conditions to add or remove a link such that the PageRank of the set $\mathcal{I}$ increases. Unlike the previous results, the particular structure of the matrix S is used to derive the next propositions. The key element is that when you add/remove a link (i,j), you decrease/increase the weights of the links $(i,k) \in \mathcal{L}$ with $k \neq j$ such that their proportionalities are preserved.

However, that redistribution of the weights is satisfied if we consider the matrix P defined in Eq. (2.4) of Chapter 2. From that particular construction, we have that every link (i,j) has a given weight w_{ij} and the vector $\boldsymbol{\delta}$ defined in Eq. (5.4) for S becomes

$$\boldsymbol{\delta} = \frac{1}{\sum_{j \,\widetilde{\leftarrow}\, i} w_{ij}} \sum_{j \,\widetilde{\leftarrow}\, i} w_{ij}\, \boldsymbol{e}_j - \frac{1}{\sum_{j \leftarrow i} w_{ij}} \sum_{j \leftarrow i} w_{ij}\, \boldsymbol{e}_j. \tag{5.7}$$

We will see that the rest of the results remain valid if we replace the matrix S by any matrix P built according to Eq. (2.4).

Adding a link. The following Proposition 4 shows how to add a new link (i,j) starting from a given node i in order to increase the PageRank of the set $\mathcal{I}$. The PageRank of $\mathcal{I}$ increases as soon as a node $i \in \mathcal{I}$ adds a link to a node j with a larger or equal expected number of visits to $\mathcal{I}$ before zapping.

Proposition 4 (Adding a link)**.** *Let $i \in \mathcal{I}$, let $j \in \mathcal{N}$ and let $\widetilde{\mathcal{L}} = \mathcal{L} \cup \{(i,j)\}$. If $(i,j) \notin \mathcal{L}$ and $\boldsymbol{v}_i \leq \boldsymbol{v}_j$, then*

$$\widetilde{\boldsymbol{\pi}}^T \boldsymbol{e}_{\mathcal{I}} \geq \boldsymbol{\pi}^T \boldsymbol{e}_{\mathcal{I}}$$

with equality if and only if the node i does not have access to $\overline{\mathcal{I}}$.

Proof: If $i \in \mathcal{I}$ and $j \in \mathcal{N}$ are such that $(i,j) \notin \mathcal{L}$ and $\boldsymbol{v}_i \leq \boldsymbol{v}_j$. Then

$$1 + c \sum_{k \leftarrow i} \frac{\boldsymbol{v}_k}{\boldsymbol{d}_i} = \boldsymbol{v}_i \leq 1 + c\boldsymbol{v}_i \leq 1 + c\boldsymbol{v}_j,$$

with equality if and only if i does not have access to $\overline{\mathcal{I}}$ by Lemma 2(b). Let $\widetilde{\mathcal{L}} = \mathcal{L} \cup \{(i,j)\}$. Then

$$\boldsymbol{\delta}^T \boldsymbol{v} = \frac{1}{\boldsymbol{d}_i + 1} \left(\boldsymbol{v}_j - \sum_{k \leftarrow i} \frac{\boldsymbol{v}_k}{\boldsymbol{d}_i} \right) \geq 0,$$

with equality if and only if i does not have access to $\overline{\mathcal{I}}$. The conclusion follows from Corollary 2. ■

Using P from Eq. (2.4) instead of S, the proof is similar except that we use the replacement in (5.3) and $\boldsymbol{\delta}^T \boldsymbol{v}$ must be derived from Eq. (5.7). We remark that the condition is then that $\boldsymbol{v}_j$ must be greater than the weighted average $\frac{1}{\sum_{k \leftarrow i} w_{ik}} \sum_{k \leftarrow i} w_{ik}\, \boldsymbol{v}_k$ for the children of i.

Removing a link. Let us see how to remove a link (i,j) starting from a given node i in order to increase the PageRank of the set $\mathcal{I}$. If a node $i \in \mathcal{N}$ removes a link to its worst child from the point of view of the expected number of visits to $\mathcal{I}$ before zapping, then the PageRank of $\mathcal{I}$ increases.

Proposition 5 (Removing a link)**.** *Let $i \in \mathcal{N}$, let $j \in \operatorname{argmin}_{k \leftarrow i} \boldsymbol{v}_k$, and let $\widetilde{\mathcal{L}} = \mathcal{L} \setminus \{(i,j)\}$. Then*

$$\widetilde{\boldsymbol{\pi}}^T \boldsymbol{e}_{\mathcal{I}} \geq \boldsymbol{\pi}^T \boldsymbol{e}_{\mathcal{I}}$$

with equality if and only if $\boldsymbol{v}_k = \boldsymbol{v}_j$ for every k such that $(i,k) \in \mathcal{L}$.

Proof: Let $i \in \mathcal{N}$ and let $j \in \operatorname{argmin}_{k \leftarrow i} \boldsymbol{v}_k$. Let $\widetilde{\mathcal{L}} = \mathcal{L} \setminus \{(i,j)\}$. Then

$$\boldsymbol{\delta}^T \boldsymbol{v} = \sum_{k \leftarrow i} \frac{\boldsymbol{v}_k - \boldsymbol{v}_j}{\boldsymbol{d}_i(\boldsymbol{d}_i - 1)} \geq 0,$$

with equality if and only if $\boldsymbol{v}_k = \boldsymbol{v}_j$ for all $k \leftarrow i$. The conclusion follows by Corollary 2. ■

The proof for P from Eq. (2.4) instead of S can be established using the same remarks made after the proof of Proposition 4.

In order to increase the PageRank of $\mathcal{I}$ with a new link (i, j), Proposition 4 only requires that $\boldsymbol{v}_j \leq \boldsymbol{v}_i$. On the other hand, Proposition 5 requires that $\boldsymbol{v}_j = \min_{k \leftarrow i} \boldsymbol{v}_k$ in order to increase the PageRank of $\mathcal{I}$ by deleting link (i, j). One could wonder whether or not this condition could be weakened to $\boldsymbol{v}_j < \boldsymbol{v}_i$, so as to have symmetric conditions for the addition or deletion of links. In fact, this can not be done as shown in the following example.

Example 2. Let us see by an example that the condition $j \in \operatorname{argmin}_{k \leftarrow i} \boldsymbol{v}_k$ in Proposition 5 can not be weakened to $\boldsymbol{v}_j < \boldsymbol{v}_i$. Consider the network in Fig. 5.3 and take a damping factor $c = 0.85$. Let $\mathcal{I} = \{1, 2, 3\}$. We have

$$\boldsymbol{v}_1 = 2.63 > \boldsymbol{v}_2 = 2.303 > \boldsymbol{v}_3 = 1.533.$$

As ensured by Proposition 5, if we remove the link $(1, 3)$, the PageRank of $\mathcal{I}$ increases (e.g. from 0.199 to 0.22 with a uniform personalization vector $\boldsymbol{z} = \frac{1}{n}\mathbf{1}$), since $3 \in \operatorname{argmin}_{k \leftarrow 1} \boldsymbol{v}_k$. But, if we remove instead the link $(1, 2)$, the PageRank of $\mathcal{I}$ decreases (from 0.199 to 0.179 with $\boldsymbol{z}$ a uniform personalization vector) even if $\boldsymbol{v}_2 < \boldsymbol{v}_1$.

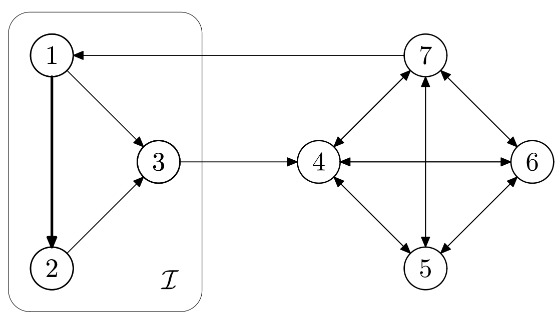


Figure 5.3: For $\mathcal{I} = \{1, 2, 3\}$, removing link $(1, 2)$ gives $\tilde{\boldsymbol{\pi}}^T \boldsymbol{e}_{\mathcal{I}} < \boldsymbol{\pi}^T \boldsymbol{e}_{\mathcal{I}}$, even if $\boldsymbol{v}_1 > \boldsymbol{v}_2$ (see Example 2).

Remark. Let us note that, if the node i does not have access to the set $\overline{\mathcal{I}}$, then for every *deletion* of a link starting from i, the PageRank of $\mathcal{I}$ will not be modified. Indeed, in this case $\boldsymbol{\delta}^T \boldsymbol{v} = 0$ since by Lemma 2(b), $\boldsymbol{v}_j = \frac{1}{1-c}$ for every $j \leftarrow i$.

Basic absorbing network. Now, let us introduce briefly the notion of basic absorbing network (see Chapter III about absorbing Markov chains in Kemeny and Snell's book [43]).

For a given network $\mathcal{G} = (\mathcal{N}, \mathcal{L})$ and a specified subset of nodes $\mathcal{I} \subseteq \mathcal{N}$, the *basic absorbing network* is the network $\mathcal{G}_0 = (\mathcal{N}, \mathcal{L}^0)$ defined by $\mathcal{L}^0_{\mathrm{out}(\mathcal{I})} = \emptyset$, $\mathcal{L}^0_{\mathcal{I}} = \{(i,i)\colon i \in \mathcal{I}\}$, $\mathcal{L}^0_{\mathrm{in}(\mathcal{I})} = \mathcal{L}_{\mathrm{in}(\mathcal{I})}$ and $\mathcal{L}^0_{\overline{\mathcal{I}}} = \mathcal{L}_{\overline{\mathcal{I}}}$. In other words, the basic absorbing network $(\mathcal{N}, \mathcal{L}^0)$ is a network constructed from $(\mathcal{N}, \mathcal{L})$, keeping the same sets of external inlinks and external links $\mathcal{L}_{\mathrm{in}(\mathcal{I})}, \mathcal{L}_{\overline{\mathcal{I}}}$, removing the external outlinks $\mathcal{L}_{\mathrm{out}(\mathcal{I})}$ and changing the internal link structure $\mathcal{L}_{\mathcal{I}}$ in order to have only self-links for nodes of $\mathcal{I}$.

Like in the previous subsection, every item corresponding to the basic absorbing network will have a zero symbol. For instance, we will write $\boldsymbol{\pi}_0$ for the PageRank vector corresponding to the basic absorbing network and $\mathcal{V}_0 = \operatorname{argmax}_{j \in \overline{\mathcal{I}}} [(I - cS_0)^{-1} \boldsymbol{e}_{\mathcal{I}}]_j$.

Proposition 6 (PageRank for a basic absorbing network)**.** *Let a network be defined by a set of links $\mathcal{L}$ and let $\mathcal{I} \subseteq \mathcal{N}$. Then*

$$\boldsymbol{\pi}^T \boldsymbol{e}_{\mathcal{I}} \leq \boldsymbol{\pi}_0^T \boldsymbol{e}_{\mathcal{I}},$$

with equality if and only if $\mathcal{L}_{\mathrm{out}(\mathcal{I})} = \emptyset$.

Proof: After a permutation of the indices, Eq. (5.1) can be written as

$$\begin{pmatrix} I - cS_{\mathcal{I}} & -cS_{\mathrm{out}(\mathcal{I})} \\ -cS_{\mathrm{in}(\mathcal{I})} & I - cS_{\overline{\mathcal{I}}} \end{pmatrix} \begin{pmatrix} \boldsymbol{v}_{\mathcal{I}} \\ \boldsymbol{v}_{\overline{\mathcal{I}}} \end{pmatrix} = \begin{pmatrix} \mathbf{1} \\ 0 \end{pmatrix},$$

so we get

$$\boldsymbol{v} = \begin{pmatrix} \boldsymbol{v}_{\mathcal{I}} \\ c(I - cS_{\overline{\mathcal{I}}})^{-1} S_{\mathrm{in}(\mathcal{I})} \boldsymbol{v}_{\mathcal{I}} \end{pmatrix}. \tag{5.8}$$

By Lemma 2(b) and since $(I - cS_{\overline{\mathcal{I}}})^{-1}$ is a nonnegative matrix (see for instance the chapter on M-matrices in Berman and Plemmons's book [9]), we then have

$$\boldsymbol{v} \leq \begin{pmatrix} \frac{1}{1-c} \mathbf{1} \\ \frac{c}{1-c} (I - cS_{\overline{\mathcal{I}}})^{-1} S_{\mathrm{in}(\mathcal{I})} \mathbf{1} \end{pmatrix} = \boldsymbol{v}_0,$$

with equality if and only if no node of $\mathcal{I}$ has access to $\overline{\mathcal{I}}$, that is $\mathcal{L}_{\mathrm{out}(\mathcal{I})} = \emptyset$. The conclusion now follows from Eq. (5.2) and $\boldsymbol{z} > 0$. ■

Let us finally prove a nice property of the set $\mathcal{V}$ when $\mathcal{I} = \{i\}$ is a singleton: it is independent of the outlinks of i. In particular, it can be found from the basic absorbing network.

Lemma 5. *Let a network defined by a set of links $\mathcal{L}$ and let $\mathcal{I} = \{i\}$. Then there exists an $\alpha \neq 0$ such that $(I - cS)^{-1}\mathbf{e}_i = \alpha(I - cS_0)^{-1}\mathbf{e}_i$. As a consequence,*

$$\mathcal{V} = \mathcal{V}_0.$$

Proof: Let $\mathcal{I} = \{i\}$. Since $\boldsymbol{v}_{\mathcal{I}} = \boldsymbol{v}_i$ is a scalar, it follows from Eq. (5.8) that the direction of the vector $\boldsymbol{v}$ does not depend on $\mathcal{L}_{\mathcal{I}}$ and $\mathcal{L}_{\text{out}(\mathcal{I})}$ but only on $\mathcal{L}_{\text{in}(\mathcal{I})}$ and $\mathcal{L}_{\overline{\mathcal{I}}}$. ■

Proposition 6 and Lemma 5 remain valid for any row stochastic matrix P. It suffices to replace S by P in the proofs.

5.3 Optimal linkage strategy for a website

In this section, we consider a set of nodes $\mathcal{I}$. For this set, we want to choose the sets of internal links $\mathcal{L}_{\mathcal{I}} \subseteq \mathcal{I} \times \mathcal{I}$ and external outlinks $\mathcal{L}_{\text{out}(\mathcal{I})} \subseteq \mathcal{I} \times \overline{\mathcal{I}}$ in order to maximize the PageRank score of $\mathcal{I}$, that is $\boldsymbol{\pi}^T \boldsymbol{e}_{\mathcal{I}}$.

The accessibility assumption. Let us first discuss the constraints on $\mathcal{L}$ we will consider. If we do not impose any condition on $\mathcal{L}$, the problem of maximizing $\boldsymbol{\pi}^T \boldsymbol{e}_{\mathcal{I}}$ is quite trivial. As shown by Proposition 6, you should take in this case $\mathcal{L}_{\text{out}(\mathcal{I})} = \emptyset$ and $\mathcal{L}_{\mathcal{I}}$ an arbitrary subset of $\mathcal{I} \times \mathcal{I}$ such that each node has at least one outlink. You just try to lure the random walker to your pages, not allowing him to leave $\mathcal{I}$ except by zapping according to the personalization vector. Therefore, it seems sensible to impose that $\mathcal{L}_{\text{out}(\mathcal{I})}$ *must be nonempty.*

Now, let us show that, in order to avoid trivial solutions to our maximization problem, it is not enough to assume that $\mathcal{L}_{\text{out}(\mathcal{I})}$ must be nonempty. Indeed, with this single constraint, in order to lose as few as possible visits from the random walker, you should take a unique leaking node $k \in \mathcal{I}$ (i.e., $\mathcal{L}_{\text{out}(\mathcal{I})} = \{(k, \ell)\}$ for some $\ell \in \overline{\mathcal{I}}$) and isolate it from the rest of the set $\mathcal{I}$ (i.e., $\{i \in \mathcal{I} \colon (i, k) \in \mathcal{L}_{\mathcal{I}}\} = \emptyset$).

Moreover, it seems reasonable to imagine that Google penalizes (or at least tries to penalize) such behavior in the context of spam alliances [30].

All this discussion leads us to make the following assumption.

Assumption A (Accessibility)**.** Every node of $\mathcal{I}$ has access to at least one node of $\overline{\mathcal{I}}$.

Presentation of the three theorems. Let us explain the basic ideas we will use in order to determine an optimal linkage strategy for a set of webpages $\mathcal{I}$. We determine some forbidden patterns for an optimal linkage strategy and deduce the only possible structure an optimal strategy can have. In other words, we assume that we have a configuration which gives an optimal PageRank $\boldsymbol{\pi}^T \boldsymbol{e}_{\mathcal{I}}$. Then we prove that if some particular pattern appeared in this optimal structure, then we could construct another network for which the PageRank $\widetilde{\boldsymbol{\pi}}^T \boldsymbol{e}_{\mathcal{I}}$ is strictly higher than $\boldsymbol{\pi}^T \boldsymbol{e}_{\mathcal{I}}$.

We will firstly determine the shape of an optimal external outlink structure $\mathcal{L}_{\mathrm{out}(\mathcal{I})}$, when the internal link structure $\mathcal{L}_{\mathcal{I}}$ is given, in Theorem 4. Then, given the external outlink structure $\mathcal{L}_{\mathrm{out}(\mathcal{I})}$ we will determine the possible optimal internal link structure $\mathcal{L}_{\mathcal{I}}$ in Theorem 5. Finally, we will put both results together in Theorem 6 in order to get the general shape of an optimal linkage strategy for a set $\mathcal{I}$ when $\mathcal{L}_{\mathrm{in}(\mathcal{I})}$ and $\mathcal{L}_{\overline{\mathcal{I}}}$ are given.

The three theorems remain valid if we replace S by any stochastic matrix P built from Eq. (2.4). This is because the proofs are based on the previous lemmas, propositions and theorems that are valid for P. Therefore, the proofs for that more general case are verbatim the same.

Proofs of this section will be illustrated by several figures for which we take the following drawing convention.

Convention. When nodes are drawn from left to right on the same horizontal line, they are arranged by decreasing value of $\boldsymbol{v}_j$. Links are represented by continuous links and paths by dashed links.

Optimal outlink structure. The first result of this section concerns the optimal *outlink* structure $\mathcal{L}_{\mathrm{out}(\mathcal{I})}$ for the set $\mathcal{I}$, while its internal structure $\mathcal{L}_{\mathcal{I}}$ is given. An example of optimal outlink structure is given after the theorem.

Theorem 4 (Optimal outlink structure)**.** *Let $\mathcal{L}_{\mathcal{I}}$, $\mathcal{L}_{\mathrm{in}(\mathcal{I})}$ and $\mathcal{L}_{\overline{\mathcal{I}}}$ be given. Let $\mathcal{F}_1, \dots, \mathcal{F}_r$ be the final classes of the subnetwork $(\mathcal{I}, \mathcal{L}_{\mathcal{I}})$. Let $\mathcal{L}_{\mathrm{out}(\mathcal{I})}$ be such that the PageRank $\boldsymbol{\pi}^T \boldsymbol{e}_{\mathcal{I}}$ is maximal under Assumption A. Then $\mathcal{L}_{\mathrm{out}(\mathcal{I})}$ has the following structure:*

$$\mathcal{L}_{\mathrm{out}(\mathcal{I})} = \mathcal{L}_{\mathrm{out}(\mathcal{F}_1)} \cup \cdots \cup \mathcal{L}_{\mathrm{out}(\mathcal{F}_r)},$$

where for every $s = 1, \dots, r$,

$$\mathcal{L}_{\mathrm{out}(\mathcal{F}_s)} \subseteq \{(i,j) \colon i \in \operatorname*{argmin}_{k \in \mathcal{F}_s} \boldsymbol{v}_k \textit{ and } j \in \mathcal{V}\}.$$

Moreover for every $s = 1, \dots, r$, if $\mathcal{L}_{\mathcal{F}_s} \neq \emptyset$, then $|\mathcal{L}_{\mathrm{out}(\mathcal{F}_s)}| = 1$.

Proof: Let $\mathcal{L}_{\mathcal{I}}$, $\mathcal{L}_{\mathrm{in}(\mathcal{I})}$ and $\mathcal{L}_{\overline{\mathcal{I}}}$ be given. Suppose $\mathcal{L}_{\mathrm{out}(\mathcal{I})}$ is such that $\boldsymbol{\pi}^T \boldsymbol{e}_{\mathcal{I}}$ is maximal under Assumption A.

We will determine the possible leaking nodes in $\mathcal{I}$ by analyzing three different cases: final classes with one node, final classes with more than one node and nodes that do not belong to a final class.

Firstly, we suppose $\mathcal{F}_s = \{i\}$ for some $s = 1, \dots, r$ and $\mathcal{L}_{\mathcal{F}_s} = \emptyset$. Clearly, $i \in \operatorname{argmin}_{k \in \mathcal{F}_s} \boldsymbol{v}_k$ and from Assumption A, i must have access to $\overline{\mathcal{I}}$, that is, $\mathcal{L}_{\mathrm{out}(\mathcal{F}_s)} \neq \emptyset$. Finally, from Corollary 2 and the optimality assumption, we have $\mathcal{L}_{\mathrm{out}(\mathcal{F}_s)} \subseteq \{(i,j) \colon j \in \mathcal{V}\}$, otherwise if $\boldsymbol{v}_j < \boldsymbol{v}_\ell$ for some $\ell \in \overline{\mathcal{I}}$, then i should replace (i,j) by (i,ℓ) (see Fig. 5.4).

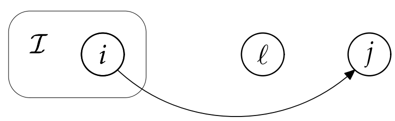


Figure 5.4: If $\boldsymbol{v}_j < \boldsymbol{v}_\ell$, then $\tilde{\boldsymbol{\pi}}^T \boldsymbol{e}_{\mathcal{I}} > \boldsymbol{\pi}^T \boldsymbol{e}_{\mathcal{I}}$ with $\widetilde{\mathcal{L}}_{\mathrm{out}(\mathcal{I})} = \mathcal{L}_{\mathrm{out}(\mathcal{I})} \cup \{(i,\ell)\} \setminus \{(i,j)\}$.

Secondly, we consider $|\mathcal{F}_s| > 1$ for some $s = 1, \dots, r$ and $\mathcal{L}_{\mathcal{F}_s} \neq \emptyset$. By Lemma 4, there is a decreasing path $(i_0, \dots, i, j)$ in $\mathcal{F}_s$ with $i \in \mathcal{I}$, $j \in \overline{\mathcal{I}}$ and such that $j \in \operatorname{argmin}_{k \leftarrow i} \boldsymbol{v}_k$. Suppose by contradiction that the node i would keep its access to $\overline{\mathcal{I}}$ if we took $\widetilde{\mathcal{L}}_{\mathrm{out}(\mathcal{I})} = \mathcal{L}_{\mathrm{out}(\mathcal{I})} \setminus \{(i,j)\}$ instead of $\mathcal{L}_{\mathrm{out}(\mathcal{I})}$. Then, by Proposition 5, considering $\widetilde{\mathcal{L}}_{\mathrm{out}(\mathcal{I})}$ instead of $\mathcal{L}_{\mathrm{out}(\mathcal{I})}$ would increase strictly the PageRank of $\mathcal{I}$ while Assumption A remains satisfied (see Fig. 5.5). This would contradict the opti-

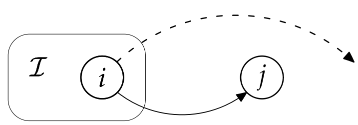


Figure 5.5: If $\boldsymbol{v}_j = \min_{k \leftarrow i} \boldsymbol{v}_k$ and i has another access to $\overline{\mathcal{I}}$, then $\tilde{\boldsymbol{\pi}}^T \boldsymbol{e}_{\mathcal{I}} > \boldsymbol{\pi}^T \boldsymbol{e}_{\mathcal{I}}$ with $\widetilde{\mathcal{L}}_{\mathrm{out}(\mathcal{I})} = \mathcal{L}_{\mathrm{out}(\mathcal{I})} \setminus \{(i,j)\}$.

mality assumption for $\mathcal{L}_{\mathrm{out}(\mathcal{I})}$. From this, we conclude that

- the node i belongs to final class $\mathcal{F}_s$ of the subnetwork $(\mathcal{I}, \mathcal{L}_\mathcal{I})$ with $\mathcal{L}_{\mathcal{F}_s} \neq \emptyset$ for some $s = 1, \dots, r$;
- there does not exist another $\ell \in \overline{\mathcal{I}}$, $\ell \neq j$ such that $(i, \ell) \in \mathcal{L}_{\text{out}(\mathcal{I})}$;
- there does not exist another k in the same final class $\mathcal{F}_s$, $k \neq i$ such that $(k, \ell) \in \mathcal{L}_{\text{out}(\mathcal{I})}$ for some $\ell \in \overline{\mathcal{I}}$.

Since i is the only leaking node in $\mathcal{F}_s$ and terminates the decreasing path in $\mathcal{I}$, we have

$$i \in \operatorname*{argmin}_{k \in \mathcal{F}_s} \boldsymbol{v}_k.$$

Moreover, by Corollary 2 and the optimality assumption, we have $j \in \mathcal{V}$ (see Fig. 5.4).

Let us now notice that

$$\max_{k \in \overline{\mathcal{I}}} \boldsymbol{v}_k < \min_{k \in \mathcal{I}} \boldsymbol{v}_k. \tag{5.9}$$

Indeed, with $i \in \operatorname{argmin}_{k \in \mathcal{I}} \boldsymbol{v}_k$, we are in one of the two cases analyzed above for which we have seen that $\boldsymbol{v}_i > \boldsymbol{v}_j = \operatorname{argmax}_{k \in \overline{\mathcal{I}}} \boldsymbol{v}_k$.

Finally, consider a node $i \in \mathcal{I}$ that does not belong to any of the final classes of the subnetwork $(\mathcal{I}, \mathcal{L}_\mathcal{I})$. Suppose by contradiction that there exists $j \in \overline{\mathcal{I}}$ such that $(i, j) \in \mathcal{L}_{\text{out}(\mathcal{I})}$. Let $\ell \in \operatorname{argmin}_{k \leftarrow i} \boldsymbol{v}_k$. Then it follows from inequality (5.9) that $\ell \in \overline{\mathcal{I}}$. But the same argument as above shows that the link $(i, \ell) \in \mathcal{L}_{\text{out}(\mathcal{I})}$ must be removed since $\mathcal{L}_{\text{out}(\mathcal{I})}$ is supposed to be optimal (see Fig. 5.5 again). So, there does not exist $j \in \overline{\mathcal{I}}$ such that $(i, j) \in \mathcal{L}_{\text{out}(\mathcal{I})}$ for a node $i \in \mathcal{I}$ which does not belong to any of the final classes $\mathcal{F}_1, \dots, \mathcal{F}_r$. ■

Example 3. Let us consider the network given in Fig. 5.6. The internal link structure $\mathcal{L}_\mathcal{I}$, as well as $\mathcal{L}_{\text{in}(\mathcal{I})}$ and $\mathcal{L}_{\overline{\mathcal{I}}}$ are given. The subnetwork $(\mathcal{I}, \mathcal{L}_\mathcal{I})$ has two final classes $\mathcal{F}_1$ and $\mathcal{F}_2$. With $c = 0.85$ and $\boldsymbol{z}$ the uniform probability vector, this configuration has six optimal outlink structures (one of these solutions is represented by bold links in Fig. 5.6). Each one can be written as $\mathcal{L}_{\text{out}(\mathcal{I})} = \mathcal{L}_{\text{out}(\mathcal{F}_1)} \cup \mathcal{L}_{\text{out}(\mathcal{F}_2)}$, with $\mathcal{L}_{\text{out}(\mathcal{F}_1)} = \{(4, 6)\}$ or $\mathcal{L}_{\text{out}(\mathcal{F}_1)} = \{(4, 7)\}$ and $\emptyset \neq \mathcal{L}_{\text{out}(\mathcal{F}_2)} \subseteq \{(5, 6), (5, 7)\}$. Indeed, since $\mathcal{L}_{\mathcal{F}_1} \neq \emptyset$, as stated by Theorem 4, the final class $\mathcal{F}_1$ has exactly one external outlink in every optimal outlink structure. On the other hand, the final class $\mathcal{F}_2$ may have several external outlinks, since it is composed of a unique node and moreover this node does not have a self-link. Note that $\mathcal{V} = \{6, 7\}$ in each of these six optimal configurations, but this set $\mathcal{V}$ can not be determined a priori since it depends on the chosen outlink structure.

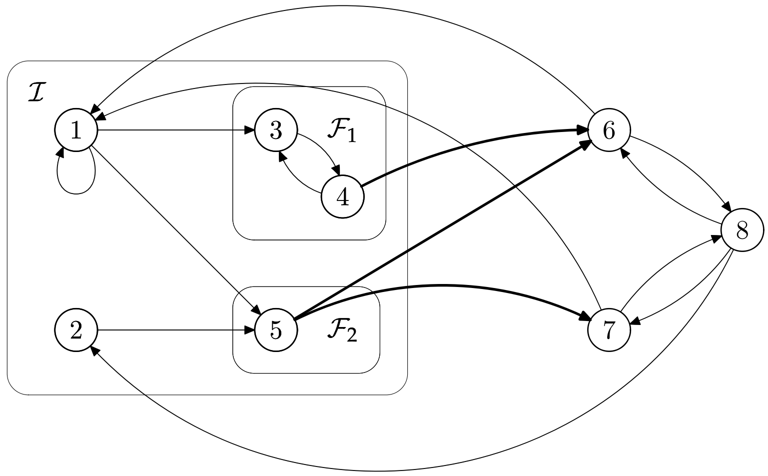


Figure 5.6: Bold links represent one of the six optimal *outlink* structures for this configuration with two final classes (see Example 3).

Optimal internal link structure. Let us determine the optimal *internal* link structure $\mathcal{L}_\mathcal{I}$ for the set $\mathcal{I}$, while its outlink structure $\mathcal{L}_{\mathrm{out}(\mathcal{I})}$ is given. Examples of optimal internal structure are given after the proof of the theorem.

Theorem 5 (Optimal internal link structure)**.** *Let $\mathcal{L}_{\mathrm{out}(\mathcal{I})}$, $\mathcal{L}_{\mathrm{in}(\mathcal{I})}$ and $\mathcal{L}_{\overline{\mathcal{I}}}$ be given. Let $\mathcal{Q} = \{i \in \mathcal{I} \colon (i,j) \in \mathcal{L}_{\mathrm{out}(\mathcal{I})}$ for some $j \in \overline{\mathcal{I}}\}$ be the set of leaking nodes of $\mathcal{I}$ and let $n_\mathcal{Q} = |\mathcal{Q}|$ be the number of leaking nodes. Let $\mathcal{L}_\mathcal{I}$ such that the PageRank $\boldsymbol{\pi}^T \boldsymbol{e}_\mathcal{I}$ is maximal under Assumption A. Then there exists a permutation of the indices such that $\mathcal{I} = \{1, 2, \dots, n_\mathcal{I}\}$, $\mathcal{Q} = \{n_\mathcal{I} - n_\mathcal{Q} + 1, \dots, n_\mathcal{I}\}$,*

$$\boldsymbol{v}_1 > \cdots > \boldsymbol{v}_{n_\mathcal{I} - n_\mathcal{Q}} > \boldsymbol{v}_{n_\mathcal{I} - n_\mathcal{Q} + 1} \geq \cdots \geq \boldsymbol{v}_{n_\mathcal{I}},$$

and $\mathcal{L}_\mathcal{I}$ has the following structure:

$$\mathcal{L}_\mathcal{I}^L \subseteq \mathcal{L}_\mathcal{I} \subseteq \mathcal{L}_\mathcal{I}^U,$$

where

$$\begin{aligned} \mathcal{L}_\mathcal{I}^L &= \{(i,j) \in \mathcal{I} \times \mathcal{I} \colon j \leq i\} \cup \{(i,j) \in (\mathcal{I} \setminus \mathcal{Q}) \times \mathcal{I} \colon j = i + 1\}, \\ \mathcal{L}_\mathcal{I}^U &= \mathcal{L}_\mathcal{I}^L \cup \{(i,j) \in \mathcal{Q} \times \mathcal{Q} \colon i < j\}. \end{aligned}$$

Proof: Let $\mathcal{L}_{\mathrm{out}(\mathcal{I})}$, $\mathcal{L}_{\mathrm{in}(\mathcal{I})}$ and $\mathcal{L}_{\overline{\mathcal{I}}}$ be given. Suppose $\mathcal{L}_{\mathcal{I}}$ is such that $\boldsymbol{\pi}^T \boldsymbol{e}_{\mathcal{I}}$ is maximal under Assumption A.

Firstly, by Proposition 4 and since every node of $\mathcal{I}$ has access to $\overline{\mathcal{I}}$, every node $i \in \mathcal{I}$ links to every node $j \in \mathcal{I}$ such that $\boldsymbol{v}_j \geq \boldsymbol{v}_i$ (see Fig. 5.7), that is

$$\{(i,j) \in \mathcal{L}_{\mathcal{I}} \colon \boldsymbol{v}_i \leq \boldsymbol{v}_j\} = \{(i,j) \in \mathcal{I} \times \mathcal{I} \colon \boldsymbol{v}_i \leq \boldsymbol{v}_j\}. \tag{5.10}$$

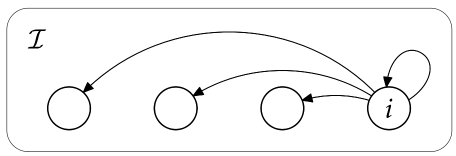


Figure 5.7: Every $i \in \mathcal{I}$ must link to every $j \in \mathcal{I}$ with $\boldsymbol{v}_j \geq \boldsymbol{v}_i$.

Secondly, let $(k,i) \in \mathcal{L}_{\mathcal{I}}$ such that $k \neq i$ and $k \in \mathcal{I} \setminus \mathcal{Q}$. Let us prove that, if the node i has access to $\overline{\mathcal{I}}$ by a path $(i, i_1, \dots, i_s)$ such that $i_j \neq k$ for all $j = 1, \dots, s$ and $i_s \in \overline{\mathcal{I}}$, then $\boldsymbol{v}_i < \boldsymbol{v}_k$ (see Fig. 5.8). Indeed, if we

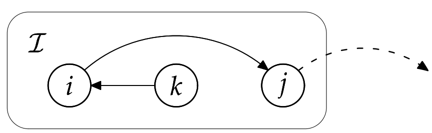


Figure 5.8: The node i can not have access to $\overline{\mathcal{I}}$ without crossing k since in this case we should then have $\boldsymbol{v}_i < \boldsymbol{v}_k$.

had $\boldsymbol{v}_k \leq \boldsymbol{v}_i$ then, by Lemma 2(c), there would exist $\ell \in \mathcal{I}$ such that $(k,\ell) \in \mathcal{L}_{\mathcal{I}}$ and $\boldsymbol{v}_\ell = \min_{j \leftarrow k} \boldsymbol{v}_j < \boldsymbol{v}_i$. But, with $\widetilde{\mathcal{L}}_{\mathcal{I}} = \mathcal{L}_{\mathcal{I}} \setminus \{(k,\ell)\}$, we would have $\widetilde{\boldsymbol{\pi}}^T \boldsymbol{e}_{\mathcal{I}} > \boldsymbol{\pi}^T \boldsymbol{e}_{\mathcal{I}}$ by Proposition 5 while Assumption A remains satisfied since the node k would keep access to $\overline{\mathcal{I}}$ via the node i (see Fig. 5.9). That contradicts the optimality assumption. This leads us to the conclusion that $\boldsymbol{v}_k > \boldsymbol{v}_i$ for every $k \in \mathcal{I} \setminus \mathcal{Q}$ and $i \in \mathcal{Q}$. Moreover $\boldsymbol{v}_i \neq \boldsymbol{v}_k$ for every $i, k \in \mathcal{I} \setminus \mathcal{Q}$, $i \neq k$. Indeed, if we had $\boldsymbol{v}_i = \boldsymbol{v}_k$, then $(k,i) \in \mathcal{L}_{\mathcal{I}}$ by (5.10) while by Lemma 4, the node i would have access to $\overline{\mathcal{I}}$ by a path independent from k. So we should have $\boldsymbol{v}_i < \boldsymbol{v}_k$.

We conclude from this that we can relabel the nodes of $\mathcal{N}$ such that $\mathcal{I} = \{1, 2, \dots n_{\mathcal{I}}\}$, $\mathcal{Q} = \{n_{\mathcal{I}} - n_{\mathcal{Q}} + 1, \dots, n_{\mathcal{I}}\}$ and

$$\boldsymbol{v}_1 > \boldsymbol{v}_2 > \cdots > \boldsymbol{v}_{n_{\mathcal{I}} - n_{\mathcal{Q}}} > \boldsymbol{v}_{n_{\mathcal{I}} - n_{\mathcal{Q}} + 1} \geq \cdots \geq \boldsymbol{v}_{n_{\mathcal{I}}}. \tag{5.11}$$

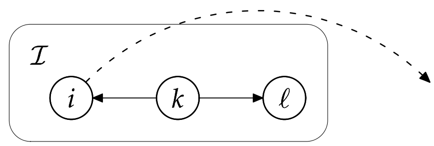


Figure 5.9: If $\boldsymbol{v}_\ell = \min_{j \leftarrow k} \boldsymbol{v}_j$, then $\tilde{\boldsymbol{\pi}}^T \boldsymbol{e}_{\mathcal{I}} > \boldsymbol{\pi}^T \boldsymbol{e}_{\mathcal{I}}$ with $\widetilde{\mathcal{L}}_{\mathrm{out}(\mathcal{I})} = \mathcal{L}_{\mathrm{out}(\mathcal{I})} \setminus \{(k,\ell)\}$.

It follows also that, for $i \in \mathcal{I} \setminus \mathcal{Q}$ and $j > i$, $(i,j) \in \mathcal{L}_{\mathcal{I}}$ if and only if $j = i+1$. Indeed, suppose first $i < n_{\mathcal{I}} - n_{\mathcal{Q}}$. Then, we cannot have $(i,j) \in \mathcal{L}_{\mathcal{I}}$ with $j > i+1$ since in this case we would contradict the ordering of the nodes given by Eq. (5.11) (see Fig. 5.8 again with $k = i+1$ and remember that by Lemma 4, node j has access to $\overline{\mathcal{I}}$ by a decreasing path). Moreover, node i must link to some node $j > i$ in order to satisfy Assumption A, so $(i, i+1)$ must belong to $\mathcal{L}_{\mathcal{I}}$. Now, consider the case $i = n_{\mathcal{I}} - n_{\mathcal{Q}}$. Suppose we had $(i,j) \in \mathcal{L}_{\mathcal{I}}$ with $j > i+1$. Let us first note that there can not exist two or more different links (i,ℓ) with $\ell \in \mathcal{Q}$ since in this case we could remove one of these links and increase strictly the PageRank of the set $\mathcal{I}$. If $\boldsymbol{v}_j = \boldsymbol{v}_{i+1}$, we could relabel the nodes by permuting these two indices. If $\boldsymbol{v}_j < \boldsymbol{v}_{i+1}$, then with $\widetilde{\mathcal{L}}_{\mathcal{I}} = \mathcal{L}_{\mathcal{I}} \cup \{(i,i+1)\} \setminus \{(i,j)\}$, we would have $\tilde{\boldsymbol{\pi}}^T \boldsymbol{e}_{\mathcal{I}} > \boldsymbol{\pi}^T \boldsymbol{e}_{\mathcal{I}}$ by Corollary 2 while Assumption A remains satisfied since node i would keep access to $\overline{\mathcal{I}}$ via node $i+1$. That contradicts the optimality assumption. So we have proved that

$$\{(i,j) \in \mathcal{L}_{\mathcal{I}} \colon i < j \text{ and } i \in \mathcal{I} \setminus \mathcal{Q}\} = \{(i,i+1) \colon i \in \mathcal{I} \setminus \mathcal{Q}\}. \quad (5.12)$$

Thirdly, it is obvious that

$$\{(i,j) \in \mathcal{L}_{\mathcal{I}} \colon i < j \text{ and } i \in \mathcal{Q}\} \subseteq \{(i,j) \in \mathcal{Q} \times \mathcal{Q} \colon i < j\}. \quad (5.13)$$

The announced structure for a set $\mathcal{L}_{\mathcal{I}}$ giving a maximal PageRank score $\boldsymbol{\pi}^T \boldsymbol{e}_{\mathcal{I}}$ under Assumption A now follows directly from Eq. (5.10), (5.12) and (5.13). ■

Example 4. Let us consider the networks given in Fig. 5.10. For both cases, the external outlink structure $\mathcal{L}_{\mathrm{out}(\mathcal{I})}$ with two leaking nodes, as well as $\mathcal{L}_{\mathrm{in}(\mathcal{I})}$ and $\mathcal{L}_{\overline{\mathcal{I}}}$ are given. With $c = 0.85$ and $\boldsymbol{z}$ the uniform probability vector, the optimal internal link structure for configuration (a) is given by $\mathcal{L}_{\mathcal{I}} = \mathcal{L}_{\mathcal{I}}^L$, while in configuration (b) we have $\mathcal{L}_{\mathcal{I}} = \mathcal{L}_{\mathcal{I}}^U$ (bold links), with $\mathcal{L}_{\mathcal{I}}^L$ and $\mathcal{L}_{\mathcal{I}}^U$ defined in Theorem 5.

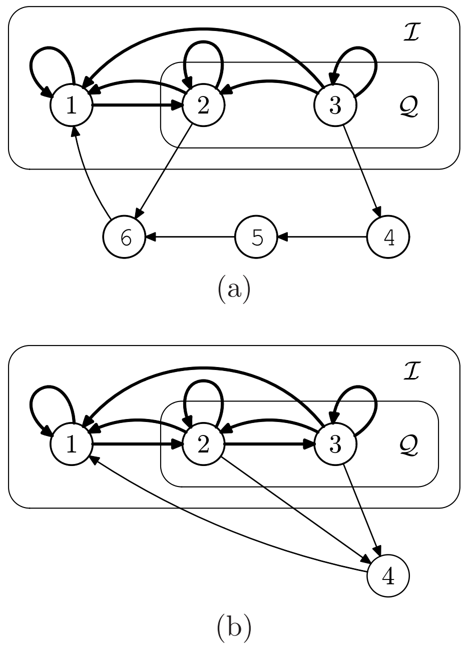


Figure 5.10: Bold links represent optimal *internal* link structures. In (a) we have $\mathcal{L}_\mathcal{I} = \mathcal{L}_\mathcal{I}^L$, while $\mathcal{L}_\mathcal{I} = \mathcal{L}_\mathcal{I}^U$ in (b).

Optimal structure. Finally, combining the optimal outlink structure and the optimal internal link structure described in Theorems 4 and 5, we find the *optimal linkage strategy* for a set of webpages. Let us note that, since we have here control on both $\mathcal{L}_\mathcal{I}$ and $\mathcal{L}_{\mathrm{out}(\mathcal{I})}$, there are no more cases of several final classes or several leaking nodes to consider. For an example of optimal link structure, see Fig. 5.1.

Theorem 6 (Optimal link structure)**.** *Let $\mathcal{L}_{\mathrm{in}(\mathcal{I})}$ and $\mathcal{L}_{\overline{\mathcal{I}}}$ be given. Let $\mathcal{L}_\mathcal{I}$ and $\mathcal{L}_{\mathrm{out}(\mathcal{I})}$ such that $\boldsymbol{\pi}^T \boldsymbol{e}_\mathcal{I}$ is maximal under Assumption A. Then there exists a permutation of the indices such that $\mathcal{I} = \{1, 2, \ldots, n_\mathcal{I}\}$,*

$$\boldsymbol{v}_1 > \cdots > \boldsymbol{v}_{n_\mathcal{I}} > \boldsymbol{v}_{n_\mathcal{I}+1} \geq \cdots \geq \boldsymbol{v}_n,$$

and $\mathcal{L}_\mathcal{I}$ and $\mathcal{L}_{\mathrm{out}(\mathcal{I})}$ have the following structure:

$$\begin{aligned} \mathcal{L}_\mathcal{I} &= \{(i,j) \in \mathcal{I} \times \mathcal{I} \colon j \leq i \text{ or } j = i+1\}, \\ \mathcal{L}_{\mathrm{out}(\mathcal{I})} &= \{(n_\mathcal{I}, n_\mathcal{I}+1)\}. \end{aligned}$$

Proof: Let $\mathcal{L}_{\mathrm{in}(\mathcal{I})}$ and $\mathcal{L}_{\overline{\mathcal{I}}}$ be given and suppose $\mathcal{L}_{\mathcal{I}}$ and $\mathcal{L}_{\mathrm{out}(\mathcal{I})}$ are such that $\boldsymbol{\pi}^T \boldsymbol{e}_{\mathcal{I}}$ is maximal under Assumption A. Let us relabel the nodes of $\mathcal{N}$ such that $\mathcal{I} = \{1, 2, \ldots, n_{\mathcal{I}}\}$ and $\boldsymbol{v}_1 \geq \cdots \geq \boldsymbol{v}_{n_{\mathcal{I}}} > \boldsymbol{v}_{n_{\mathcal{I}}+1} = \max_{j \in \overline{\mathcal{I}}} \boldsymbol{v}_j$. By Theorem 5, $(i,j) \in \mathcal{L}_{\mathcal{I}}$ for every nodes $i, j \in \mathcal{I}$ such that $j \leq i$. In particular, every node of $\mathcal{I}$ has access to node 1. Therefore, there is a unique final class $\mathcal{F}_1 \subseteq \mathcal{I}$ in the subnetwork $(\mathcal{I}, \mathcal{L}_{\mathcal{I}})$. So, by Theorem 4, $\mathcal{L}_{\mathrm{out}(\mathcal{I})} = \{(k, \ell)\}$ for some $k \in \mathcal{F}_1$ and $\ell \in \overline{\mathcal{I}}$. Without loss of generality, we can suppose that $\ell = n_{\mathcal{I}} + 1$. By Theorem 5 again, the leaking node $k = n_{\mathcal{I}}$ and therefore $(i, i+1) \in \mathcal{L}_{\mathcal{I}}$ for every node $i \in \{1, \ldots, n_{\mathcal{I}} - 1\}$. ■

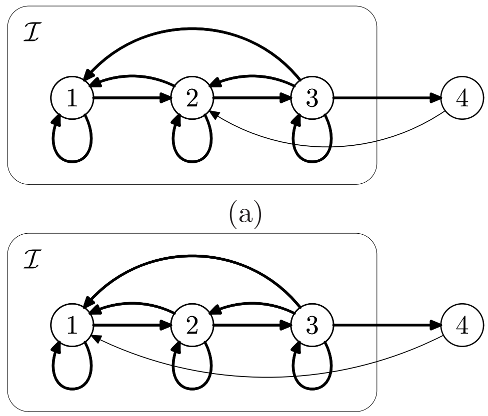
Let us note that having a structure like described in Theorem 6 is a *necessary but not sufficient* condition in order to have a maximal PageRank. To be sufficient, we should specify the order of the nodes in the particular chain so that we can control the destination of the external inlinks, see next example. If there are no external inlinks, the condition becomes equivalent.

Example 5. Let us show by an example that the network structure given in Theorem 6 is not sufficient to have a maximal PageRank. Consider for instance the networks in Fig. 5.11. Let $c = 0.85$ and a uniform personalization vector $\boldsymbol{z} = \frac{1}{n}\mathbf{1}$. Both networks have the link structure required Theorem 6 in order to have a maximal PageRank, with $\boldsymbol{v}_{(a)} = \begin{pmatrix} 6.484 & 6.42 & 6.224 & 5.457 \end{pmatrix}^T$ and $\boldsymbol{v}_{(b)} = \begin{pmatrix} 6.432 & 6.494 & 6.247 & 5.52 \end{pmatrix}^T$. But the configuration (a) is not optimal since in this case, the PageRank $\boldsymbol{\pi}_{(a)}^T \boldsymbol{e}_{\mathcal{I}} = 0.922$ is strictly less than the PageRank $\boldsymbol{\pi}_{(b)}^T \boldsymbol{e}_{\mathcal{I}} = 0.926$ obtained by the configuration (b).

5.4 Extensions and variants

Let us now present some extensions and variants of the results of the previous section. We will first emphasize the role of parents of $\mathcal{I}$. Secondly, we will briefly talk about Avrachenkov–Litvak's optimal link structure for the case where $\mathcal{I}$ is a singleton. Then we will give variants of Theorem 6 when self-links are forbidden or when a minimal number of external outlinks is required. Finally, we will make some comments of the influence of external *inlinks* on the PageRank of $\mathcal{I}$.

Like in the previous section, the results presented here remain valid if we replace S by any stochastic matrix P built from Eq. (2.4).



(a)

(b)

Figure 5.11: For $\mathcal{I} = \{1, 2, 3\}$, $c = 0.85$ and $\boldsymbol{z}$ a uniform personalization vector, the link structure in (a) is not optimal and yet it satisfies the necessary conditions of Theorem 6 (see Example 5).

Linking to parents. If some node of $\mathcal{I}$ has at least one parent in $\overline{\mathcal{I}}$ then the optimal linkage strategy for $\mathcal{I}$ is to have an internal link structure like described in Theorem 6 together with a single link to one of the parents of $\mathcal{I}$.

Corollary 3 (Necessity of linking to parents)**.** *Let $\mathcal{L}_{\mathrm{in}(\mathcal{I})} \neq \emptyset$ and $\mathcal{L}_{\overline{\mathcal{I}}}$ be given. Let $\mathcal{L}_{\mathcal{I}}$ and $\mathcal{L}_{\mathrm{out}(\mathcal{I})}$ such that $\boldsymbol{\pi}^T \boldsymbol{e}_{\mathcal{I}}$ is maximal under Assumption A. Then $\mathcal{L}_{\mathrm{out}(\mathcal{I})} = \{(i, j)\}$, for some $i \in \mathcal{I}$ and $j \in \overline{\mathcal{I}}$ such that $(j, k) \in \mathcal{L}_{\mathrm{in}(\mathcal{I})}$ for some $k \in \mathcal{I}$.*

Proof: This is a direct consequence of Lemma 3 and Theorem 6. ■

Let us nevertheless remember that not every parent of nodes of $\mathcal{I}$ will give an optimal link structure, as we have already discussed in Example 1 and we develop now.

Example 6. Let us continue Example 1. We consider the network in Fig. 5.2 as basic absorbing network for $\mathcal{I} = \{1\}$, that is $\mathcal{L}_{\mathrm{in}(\mathcal{I})}$ and $\mathcal{L}_{\overline{\mathcal{I}}}$ are given. We take $c = 0.85$ as damping factor and a uniform personalization vector $\boldsymbol{z} = \frac{1}{n}\mathbf{1}$. We have seen in Example 1 than $\mathcal{V}_0 = \{2, 3, 4\}$. Let us consider the value of the PageRank $\boldsymbol{\pi}_1$ for different sets $\mathcal{L}_{\mathcal{I}}$ and $\mathcal{L}_{\mathrm{out}(\mathcal{I})}$:

	$\mathcal{L}_{\text{out}(\mathcal{I})}$				
	$\emptyset$	$\{(1,2)\}$	$\{(1,5)\}$	$\{(1,6)\}$	$\{(1,2),(1,3)\}$
$\mathcal{L}_{\mathcal{I}} = \emptyset$	/	0.1739	0.1402	0.1392	0.1739
$\mathcal{L}_{\mathcal{I}} = \{(1,1)\}$	0.5150	0.2600	0.2204	0.2192	0.2231

The optimal linkage strategy for $\mathcal{I} = \{1\}$ is to have a self-link and a link to one of the nodes 2, 3 or 4. We note also that a link to node 6, which is a parent of node 1 provides a lower PageRank than a link to node 5, which is not parent of 1. Finally, if we suppose self-links are forbidden (see below), then the optimal linkage strategy is to link to one *or more* of the nodes $2, 3, 4$.

In the case where no node of $\mathcal{I}$ has a parent in $\overline{\mathcal{I}}$, then *every* structure like described in Theorem 6 will give an optimal link structure.

Proposition 7 (No external parent)**.** *Let $\mathcal{L}_{\text{in}(\mathcal{I})}$ and $\mathcal{L}_{\overline{\mathcal{I}}}$ be given. Suppose that $\mathcal{L}_{\text{in}(\mathcal{I})} = \emptyset$. Then the PageRank $\boldsymbol{\pi}^T \boldsymbol{e}_{\mathcal{I}}$ is maximal under Assumption A if and only if*

$$\begin{aligned} \mathcal{L}_{\mathcal{I}} &= \{(i,j) \in \mathcal{I} \times \mathcal{I} \colon j \leq i \text{ or } j = i+1\}, \\ \mathcal{L}_{\text{out}(\mathcal{I})} &= \{(n_{\mathcal{I}}, n_{\mathcal{I}}+1)\}. \end{aligned}$$

for some permutation of the indices such that $\mathcal{I} = \{1, 2, \ldots, n_{\mathcal{I}}\}$.

Proof: This follows directly from $\boldsymbol{\pi}^T \boldsymbol{e}_{\mathcal{I}} = (1-c)\boldsymbol{z}^T \boldsymbol{v}$ and the fact that, if $\mathcal{L}_{\text{in}(\mathcal{I})} = \emptyset$,

$$\boldsymbol{v} = (I - cS)^{-1} \boldsymbol{e}_{\mathcal{I}} = \begin{pmatrix} (I - cS_{\mathcal{I}})^{-1} \mathbf{1} \\ 0 \end{pmatrix},$$

after a permutation of the indices. ■

Optimal linkage strategy for a singleton. The optimal outlink structure for a single webpage has already been given by Avrachenkov and Litvak in [6]. Their result becomes a particular case of Theorem 6. Note that in the case of a single node, the possible choices for $\mathcal{L}_{\text{out}(\mathcal{I})}$ can be found a priori by considering the basic absorbing network, since $\mathcal{V} = \mathcal{V}_0$.

Corollary 4 (Optimal link structure for a single node)**.** *Let $\mathcal{I} = \{i\}$ and let $\mathcal{L}_{\text{in}(\mathcal{I})}$ and $\mathcal{L}_{\overline{\mathcal{I}}}$ be given. Then the PageRank $\boldsymbol{\pi}_i$ is maximal under Assumption A if and only if $\mathcal{L}_{\mathcal{I}} = \{(i,i)\}$ and $\mathcal{L}_{\text{out}(\mathcal{I})} = \{(i,j)\}$ for some $j \in \mathcal{V}_0$.*

Proof: This follows directly from Lemma 5 and Theorem 6. ■

Optimal linkage strategy under additional assumptions. Let us consider the problem of maximizing the PageRank $\boldsymbol{\pi}^T \boldsymbol{e}_{\mathcal{I}}$ when *self-links are forbidden.* Indeed, it seems to be often supposed that Google's PageRank algorithm does not take self-links into account. In this case, Theorem 6 can be adapted readily for the case where $|\mathcal{I}| \geq 2$. When $\mathcal{I}$ is a singleton, we must have $\mathcal{L}_{\mathcal{I}} = \emptyset$, so $\mathcal{L}_{\mathrm{out}(\mathcal{I})}$ can contain *several* links, as stated in Theorem 4.

Corollary 5 (Optimal link structure with no self-links)**.** *Suppose* $|\mathcal{I}| \geq 2$*. Let* $\mathcal{L}_{\mathrm{in}(\mathcal{I})}$ *and* $\mathcal{L}_{\overline{\mathcal{I}}}$ *be given. Let* $\mathcal{L}_{\mathcal{I}}$ *and* $\mathcal{L}_{\mathrm{out}(\mathcal{I})}$ *such that* $\boldsymbol{\pi}^T \boldsymbol{e}_{\mathcal{I}}$ *is maximal under Assumption A and assumption that there does not exist* $i \in \mathcal{I}$ *such that* $\{(i,i)\} \in \mathcal{L}_{\mathcal{I}}$*. Then there exists a permutation of the indices such that* $\mathcal{I} = \{1, 2, \ldots, n_{\mathcal{I}}\}$*,* $\boldsymbol{v}_1 > \cdots > \boldsymbol{v}_{n_{\mathcal{I}}} > \boldsymbol{v}_{n_{\mathcal{I}}+1} \geq \cdots \geq \boldsymbol{v}_n$*, and* $\mathcal{L}_{\mathcal{I}}$ *and* $\mathcal{L}_{\mathrm{out}(\mathcal{I})}$ *have the following structure:*

$$\begin{aligned} \mathcal{L}_{\mathcal{I}} &= \{(i,j) \in \mathcal{I} \times \mathcal{I} \colon j < i \text{ or } j = i+1\}, \\ \mathcal{L}_{\mathrm{out}(\mathcal{I})} &= \{(n_{\mathcal{I}}, n_{\mathcal{I}}+1)\}. \end{aligned}$$

Corollary 6 (Optimal link structure for a single node with no self-link)**.** *Suppose* $\mathcal{I} = \{i\}$*. Let* $\mathcal{L}_{\mathrm{in}(\mathcal{I})}$ *and* $\mathcal{L}_{\overline{\mathcal{I}}}$ *be given. Suppose* $\mathcal{L}_{\mathcal{I}} = \emptyset$*. Then the PageRank* $\boldsymbol{\pi}_i$ *is maximal under Assumption A if and only if* $\emptyset \neq \mathcal{L}_{\mathrm{out}(\mathcal{I})} \subseteq \mathcal{V}_0$*.*

Let us now consider the problem of maximizing the PageRank $\boldsymbol{\pi}^T \boldsymbol{e}_{\mathcal{I}}$ when *several external outlinks are required.* Then the proof of Theorem 4 can be adapted readily in order to have the following variant of Theorem 6.

Corollary 7 (Optimal link structure with several external outlinks)**.** *Let* $\mathcal{L}_{\mathrm{in}(\mathcal{I})}$ *and* $\mathcal{L}_{\overline{\mathcal{I}}}$ *be given. Let* $\mathcal{L}_{\mathcal{I}}$ *and* $\mathcal{L}_{\mathrm{out}(\mathcal{I})}$ *such that* $\boldsymbol{\pi}^T \boldsymbol{e}_{\mathcal{I}}$ *is maximal under Assumption A and assumption that* $|\mathcal{L}_{\mathrm{out}(\mathcal{I})}| \geq r$*. Then there exists a permutation of the indices such that* $\mathcal{I} = \{1, 2, \ldots, n_{\mathcal{I}}\}$*,* $\boldsymbol{v}_1 > \cdots > \boldsymbol{v}_{n_{\mathcal{I}}} > \boldsymbol{v}_{n_{\mathcal{I}}+1} \geq \cdots \geq \boldsymbol{v}_n$*, and* $\mathcal{L}_{\mathcal{I}}$ *and* $\mathcal{L}_{\mathrm{out}(\mathcal{I})}$ *have the following structure:*

$$\begin{aligned} \mathcal{L}_{\mathcal{I}} &= \{(i,j) \in \mathcal{I} \times \mathcal{I} \colon j < i \text{ or } j = i+1\}, \\ \mathcal{L}_{\mathrm{out}(\mathcal{I})} &= \{(n_{\mathcal{I}}, j_k) \colon j_k \in \mathcal{V} \text{ for } k = 1, \ldots, r\}. \end{aligned}$$

External inlinks. Finally, let us make some comments about the addition of external inlinks to the set $\mathcal{I}$. It is well known that adding an inlink to a particular page always increases the PageRank of this page [5, 38]. This can be viewed as a direct consequence of Corollary 2 and Lemma 2. The case of a set of several pages $\mathcal{I}$ is not so simple. We prove in the following theorem that, if the set $\mathcal{I}$ has a link structure as described in Theorem 6 then adding an inlink to a page of $\mathcal{I}$ from a page $j \in \overline{\mathcal{I}}$ which is *not* a parent of some node of $\mathcal{I}$ will increase the PageRank of $\mathcal{I}$. But in general, adding an inlink to some page of $\mathcal{I}$ from $\overline{\mathcal{I}}$ may *decrease* the PageRank of the set $\mathcal{I}$, as shown in Examples 7 and 8.

Theorem 7 (External inlinks)**.** *Let $\mathcal{I} \subseteq \mathcal{N}$ and let consider a network defined by a set of links $\mathcal{L}$. If*

$$\min_{i\in\mathcal{I}} \boldsymbol{v}_i > \max_{j\notin\mathcal{I}} \boldsymbol{v}_j,$$

then, for every $j \in \overline{\mathcal{I}}$ which is not a parent of $\mathcal{I}$, and for every $i \in \mathcal{I}$, the network defined by $\widetilde{\mathcal{L}} = \mathcal{L} \cup \{(j,i)\}$ gives $\widetilde{\boldsymbol{\pi}}^T \boldsymbol{e}_\mathcal{I} > \boldsymbol{\pi}^T \boldsymbol{e}_\mathcal{I}$.

Proof: This follows directly from Corollary 2. ∎

Example 7. Let us show by an example that a new external inlink is not always profitable for a set $\mathcal{I}$ in order to improve one's PageRank, even if $\mathcal{I}$ has an optimal linkage strategy. Consider for instance the network in Fig. 5.12. With $c = 0.85$ and $\boldsymbol{z}$ a uniform personalization vector, we have $\boldsymbol{\pi}^T \boldsymbol{e}_\mathcal{I} = 0.8481$. But if we consider the network defined by $\widetilde{\mathcal{L}}_{\mathrm{in}(\mathcal{I})} = \mathcal{L}_{\mathrm{in}(\mathcal{I})} \cup \{(3,2)\}$, then we have $\widetilde{\boldsymbol{\pi}}^T \boldsymbol{e}_\mathcal{I} = 0.8321 < \boldsymbol{\pi}^T \boldsymbol{e}_\mathcal{I}$.

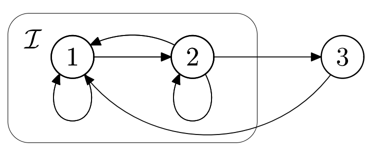


Figure 5.12: For $\mathcal{I} = \{1,2\}$, adding the external inlink $(3,2)$ gives $\widetilde{\boldsymbol{\pi}}^T \boldsymbol{e}_\mathcal{I} < \boldsymbol{\pi}^T \boldsymbol{e}_\mathcal{I}$ (see Example 7).

Example 8. A new external inlink does not always increase the PageRank of a set $\mathcal{I}$ in even if this new inlink comes from a page which is not already a parent of some node of $\mathcal{I}$. Consider for instance the network in Fig. 5.13. With $c = 0.85$ and $\boldsymbol{z}$ a uniform personalization vector, we have $\boldsymbol{\pi}^T \boldsymbol{e}_{\mathcal{I}} = 0.6$. But if we consider the network defined by $\widetilde{\mathcal{L}}_{\text{in}(\mathcal{I})} = \mathcal{L}_{\text{in}(\mathcal{I})} \cup \{(4,3)\}$, then we have $\widetilde{\boldsymbol{\pi}}^T \boldsymbol{e}_{\mathcal{I}} = 0.5897 < \boldsymbol{\pi}^T \boldsymbol{e}_{\mathcal{I}}$.

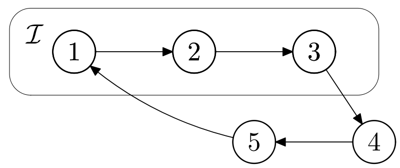


Figure 5.13: For $\mathcal{I} = \{1, 2, 3\}$, adding the external inlink $(4, 3)$ gives $\widetilde{\boldsymbol{\pi}}^T \boldsymbol{e}_{\mathcal{I}} < \boldsymbol{\pi}^T \boldsymbol{e}_{\mathcal{I}}$ (see Example 8).

5.5 Conclusions

Results. In this chapter we provide the general shape of an optimal link structure for a website in order to maximize one's PageRank or its variant where links have different weights. This structure with a forward chain and every possible backward link may be not intuitive. To our knowledge, it has never been mentioned, while topologies like a clique, a ring or a star are considered in the literature on collusion and alliance between pages [7, 30]. Moreover, this optimal structure gives new insight into the affirmation of Bianchini et al. [10] that, in order to maximize the PageRank of a website, hyperlinks to the rest of the webnetwork "should be in pages with a small PageRank and that have many internal hyperlinks". More precisely, we have seen that the leaking pages must be chosen with respect to the mean number of visits before zapping they give to the website, rather than their PageRank.

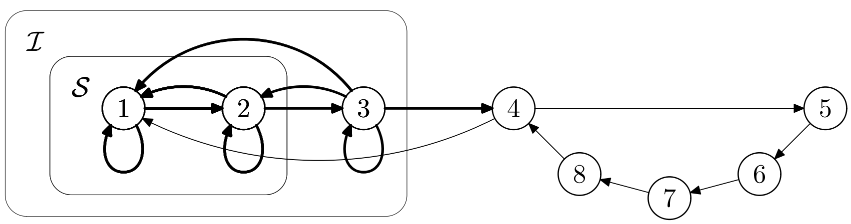
Future research. We have noticed in Example 5 that the first node of $\mathcal{I}$ in the forward chain of an optimal link structure is not necessarily *a child* of some node of $\overline{\mathcal{I}}$. In the example we gave, the personalization vector was not uniform. We wonder if this could occur with a uniform personalization vector and make the following conjecture.

Conjecture. Let $\mathcal{L}_{\text{in}(\mathcal{I})} \neq \emptyset$ and $\mathcal{L}_{\overline{\mathcal{I}}}$ be given. Let $\mathcal{L}_{\mathcal{I}}$ and $\mathcal{L}_{\text{out}(\mathcal{I})}$ such that $\boldsymbol{\pi}^T \boldsymbol{e}_{\mathcal{I}}$ is maximal under Assumption A. If $\boldsymbol{z} = \frac{1}{n}\mathbf{1}$, then there exists $j \in \overline{\mathcal{I}}$ such that $(j, i) \in \mathcal{L}_{\text{in}(\mathcal{I})}$, where $i \in \operatorname{argmax}_k \boldsymbol{v}_k$.

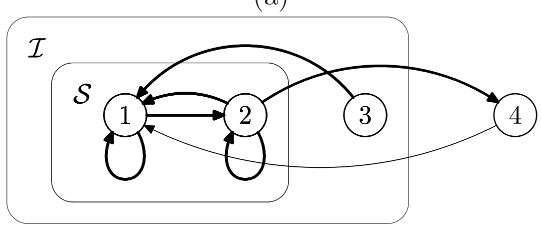
If this conjecture was true we could also ask if the node $j \in \overline{\mathcal{I}}$ such that $(j, i) \in \mathcal{L}_{\mathrm{in}(\mathcal{I})}$ where $i \in \operatorname{argmax}_k \boldsymbol{v}_k$ belongs to $\mathcal{V}$.

Another question concerns the optimal linkage strategy in order to maximize an arbitrary linear combination of the PageRanks of the nodes of $\mathcal{I}$. In particular, we could want to maximize the PageRank $\boldsymbol{\pi}^T \boldsymbol{e}_{\mathcal{S}}$ of *a target subset* $\mathcal{S} \subseteq \mathcal{I}$ by choosing $\mathcal{L}_{\mathcal{I}}$ and $\mathcal{L}_{\mathrm{out}(\mathcal{I})}$ as usual. A general shape for an optimal link structure seems difficult to find, as shown in the following example.

Example 9. Consider the networks in Fig. 5.14. In both cases, let $c = 0.85$ and $\boldsymbol{z} = \frac{1}{n}\mathbf{1}$. Let $\mathcal{I} = \{1, 2, 3\}$ and let $\mathcal{S} = \{1, 2\}$ be the target set. In the configuration (a), the optimal sets of links $\mathcal{L}_{\mathcal{I}}$ and $\mathcal{L}_{\mathrm{out}(\mathcal{I})}$ for maximizing $\boldsymbol{\pi}^T \boldsymbol{e}_{\mathcal{S}}$ has the link structure described in Theorem 6. But in (b), the optimal $\mathcal{L}_{\mathcal{I}}$ and $\mathcal{L}_{\mathrm{out}(\mathcal{I})}$ do not have this structure. Let us note nevertheless that, by Theorem 6, the subsets $\mathcal{L}_{\mathcal{S}}$ and $\mathcal{L}_{\mathrm{out}(\mathcal{S})}$ must have the link structure described in Theorem 6.



(a)



(b)

Figure 5.14: In (a) and (b), bold links represent optimal link structures for $\mathcal{I} = \{1, 2, 3\}$ with respect to a target set $\mathcal{S} = \{1, 2\}$ (see Example 9).

Chapter 6

Forbidden Nodes in Random Walks

This chapter deals with the issue of **negative links** in networks. Such links are not taken into account by most ranking methods that interpret every link as positive vote or good opinion between nodes. That is the case, for example, in eigenvector based algorithms, e.g., the PageRank, Salsa and Hits algorithms, that consider some flow through the network to define their ranking vector.

For that purpose, we introduce a natural extension of the PageRank algorithm that we label the **PageTrust algorithm**. It allows us to consider negative links used by a modified random walker to decide what nodes deserve his future visits.

We also introduce the **s-PageTrust algorithm** that is a simplified version of the PageTrust algorithm and it makes the ranking vector computable in $O(n^2)$ for sparse networks. It still has a stochastic interpretation, and simulations show close results with the original PageTrust algorithm.

Finally, we present some **properties and extensions** of the s-PageTrust algorithm. Its sensitivity to the zapping factor and another parameter – *the degree of conviction* –, its robustness when malicious users manipulate the system, its extension for flow algorithm, etc.

6.1 Introduction

Motivation. The importance of ranking methods that classify the nodes of a network by relevance is still growing more and more, especially in the context of search engines for the web. Many of them use eigenvector based techniques to extract information from the network [59]. For instance, Brin and Page's *PageRank* algorithm [85], the Kleinberg's *Hits* algorithm [51], the *Salsa* algorithm [63], and their variants [73, 81, 90] calculate dominant eigenvectors of matrices that represent the structure of that network. The nonnegativity of these matrices, i.e., the fact that all their entries are nonnegative, offers a nice and intuitive interpretation of the results in terms of random walks or in terms of flow through the links of the network. Moreover, the Perron-Frobenius theorem, see Section 2.2, claims that for any nonnegative matrix, there is at least one Perron vector which is nonnegative. If the matrix is also irreducible, that Perron vector is unique. That condition explains why it becomes nontrivial to apply eigenvector based techniques when negative links are permitted. These links would be represented by negative entries in the matrix and the nonnegativity assumption then gets lost. Though nontrivial, the consideration of negative links can be of interest to refine the measures of ranking methods. Moreover such links already exist for example in the web, but they are simply not taken into account by *Google* [71].

Existing methods. A first solution, that can be found for example in [42, 89], is to zero the entries corresponding to the negative links in the matrix representing the structure of the network. In that way, negative links do not give any trust to the nodes they point to. But then a negative link or an absence of link between two nodes amounts to the same result: in both cases, the corresponding entry in the matrix is zero. The concern with that method is therefore that the rank of a node does not change after adding a negative inlink.

A second idea, proposed in [29], is to first ignore negative links and hence to satisfy the nonnegativity assumption. The obtained eigenvector gives the trust values for all nodes, and can be interpreted as some propagation of trust through the positive links of the network. Then in order to integrate negative links, one single step of propagation of distrust is applied. As a consequence, the distrust value due to a negative link given by node i is proportional to the trust value of i. It follows that highly trusted nodes have the possibility to highly decrease the trust value of other nodes. For example, it is enough for the webpage *Yahoo!*, that has a very high PageRank, to negatively point to a node, say x, to degrade x to the end of the ranking list. This can encourage malicious nodes to negatively point to other competing nodes only to decrease their ranks and hence to pass them in the ranking list.

Another alternative, not yet investigated as far as we know, is to shift the entries of the weighted matrix to make it nonnegative. We believe that this operation deserves some attention even though we expect two difficulties: the computation because the matrix can become dense and the impact of negative links on the ranks because their number can be negligible compared to the links created by the shift, that is, the entries (i, j) of the weighted matrix that were equal to zero before shifting the matrix.

We observe in practice that the simple average method remains the most common on the Web, like in the sites *eBay* [88] and *Epinions* [70]. However, we could expect from an ideal ranking method that it takes into account the global structure of the votes in the network. Moreover, it should decrease trust values of nodes that receive more and more negative links, and be robust to attackers that want to decrease trust values of competing nodes by the means of negative links. This chapter shows how a modified random walk, that we label a *trust walk*, and how a modified flow method, that we label a *trust flow*, address these problems.

Structure. The next section formally defines the PageTrust algorithm and its associated trust walk that takes into account the negative links of the network (do not confuse with the TrustRank that is an extension of the PageRank for fighting web spam [31]). Then, Section 6.3, that represents the main part of the chapter, introduces a simplified version of the PageTrust (the s-PageTrust) to make the method calculable in $O(n^2)$ for sparse networks. We also present *the degree of conviction* that is a parameter that control the effect of negative links. In particular, when it is equal to 0, we recover the PageRank algorithm that merely ignore the negative links. Section 6.4 gives illustrated discussions on the properties of the s-PageTrust: its difference with the PageTrust, its complexity, its rate of convergence, its sensitivity with respect to its parameters, its robustness against attackers, and its results for a large dataset extracted from the website Epinions. Then, Section 6.5 proposes some possible variants and extensions of the s-PageTrust algorithm including two other algorithms, namely the FlowTrust and the s-FlowTrust algorithms, that extend the previous methods for flow algorithms. We terminate by Section 6.6 that gives some concluding remarks and perspectives for future research.

6.2 The PageTrust

In this section, we introduce the notation of the chapter and the PageTrust algorithm. That algorithm calculates the steady state probabilities, represented by the PageTrust vector, of some modified random walk that we label a trust walk.

Notation. Let $\mathcal{G} = (\mathcal{N}, \mathcal{L})$ be a directed network for which we have two distinct subsets of links: the set of positive links $\mathcal{L}^+$ and the set of negative links $\mathcal{L}^-$ with $\mathcal{L}^+ \cup \mathcal{L}^- = \mathcal{L}$ and $\mathcal{L}^+ \cap \mathcal{L}^- = \emptyset$. The outdegree $\boldsymbol{d}_i$ of a node $i \in \mathcal{N}$ will be the number of positive outlinks. The negative outlinks of i point to a set of nodes that are distrusted by i. That set is denoted by

$$\mathcal{D}_i = \{k : (i,k) \in \mathcal{L}^-\},$$

and is called the *blacklist* of i. For instance, the blacklist of i in Fig. 6.1 is $\mathcal{D}_i = \{k\}$. Examples and properties in this chapter will be illustrated by several figures for which we take the following drawing convention:

Convention. The negative links in $\mathcal{L}^-$ will be represented by dotted arrows in the network $\mathcal{G}$ (see for example Fig. 6.1).

We have seen in Section 2.3 that the PageRank vector $\boldsymbol{\pi}$ is defined by

$$\begin{aligned} \boldsymbol{\pi}^T &= \boldsymbol{\pi}^T G, \\ \boldsymbol{\pi}^T \mathbf{1} &= 1, \end{aligned}$$

where $G = cS + (1-c)\mathbf{1}\boldsymbol{z}^T$ is the Google matrix and $S = [S_{ij}]_{i,j \in \mathcal{N}}$ is the scaled adjacency matrix of the network $\mathcal{G} = (\mathcal{N}, \mathcal{L}^+)$. Remember that it is assumed that *each node has at least one outlink*, i.e., the outdegree $\boldsymbol{d}_i \neq 0$ for every $i \in \mathcal{N}$. In this chapter, the vector $\boldsymbol{\pi}$, calculated from the network $\mathcal{G}$, will be a ranking vector of a method that will be clear from the context.

Trust walk. The motion of a trust walker is identical to the one of a random walker (see Section 2.4) given by the transition matrix G of Google, except that the negative links $\mathcal{L}^-$ will modify his motion.

At time t, the walker has an opinion characterized by a blacklist $\mathcal{B}_t$ containing a subset of forbidden nodes that he will avoid in his next steps. Let $(i_0, \dots, i_t)$ be the path of our walker. If $k \in \mathcal{B}_t$, it means that some node in the path distrusts k, i.e.,

$$k \in \mathcal{D}_i \text{ for some } i = i_0, \dots, i_t.$$

Then the blacklist $\mathcal{B}_t$ is updated at time $t+1$ according to the next visited node i_{t+1}: the subset of nodes negatively pointed by node i_{t+1} are added as forbidden nodes in the blacklist, i.e., $\mathcal{B}_{t+1} = \mathcal{B}_t \cup \mathcal{D}_{i_{t+1}}$. Now, when the walker is supposed to move to a forbidden node, rather than visiting it, he empties his blacklist and *jumps* from that forbidden node to any other node according to an uniform or personalized distribution given by the vector $\boldsymbol{z}$, called the zapping vector.

On the other hand, at every step, the trust walker has a probability $1-c$ to zap according to the zapping vector $\boldsymbol{z}$. In that case, we will assume that he empties his blacklist before restarting his trust walk.

Section 6.5 discusses the variants where the blacklist is not necessarily empty before jumping and zapping. Some of these variants imply minor change in the results while some other variants lead to computational issues.

Let us point out that the trust walk we just described has some similarity with the self-avoiding walk [69]. They both update a blacklist of nodes that must be avoided in the further steps. However, in the self-avoiding walk, the blacklist contains every node that already has been visited so that the walker does not visit the same node more than once. Moreover that type of walk was mainly analyzed for regular lattices and the motivation is quite different since it is used to understand the behavior of polymers and proteins.

Example 10. A trust walk is illustrated in Fig. 6.1. Let say that at time $t = 1$, the trust walker starts in node i so that he adds node k in his blacklist, i.e., $\mathcal{B}_1 = \{k\}$. He moves then, with a probability c, to

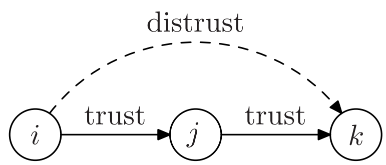


Figure 6.1: A three nodes network with $(i,j),(j,k) \in \mathcal{L}^+$ and $(i,k) \in \mathcal{L}^-$.

node j that has no negative link and therefore $\mathcal{B}_2 = \mathcal{B}_1 = \{k\}$. The second step leads the walker, with a probability c, to node k. But this node is forbidden – because distrusted by a previous encountered node –, therefore the walker jumps according to a given zapping vector $\boldsymbol{z}$ and empties his blacklist. Let us remark that the walker's blacklist clearly depends on his path.

Let us formally present the motion of a trust walker by considering the random variable

$$Y_t = (i_t, \mathcal{B}_t) \in \mathcal{N} \times \mathcal{P}(\mathcal{N})$$

that represents the state of the trust walker at time t: $i_t \in \mathcal{N}$ is the visited node at time t and $\mathcal{B}_t \in \mathcal{P}(\mathcal{N})$ is the blacklist at time t where $\mathcal{P}(\mathcal{N})$ is the power set of $\mathcal{N}$ that contains every possible blacklist. Hence, $\text{Prob}(Y_t = (i, \mathcal{B}))$ is the probability that the trust walker is in node i with blacklist $\mathcal{B}$ at time t, moreover $\text{Prob}(i_t = i)$ is the probability that the trust walker is in node i at time t, and $\text{Prob}(\mathcal{B}_t = \mathcal{B})$ is the probability that the trust walker has blacklist $\mathcal{B}$ at time t.

We remark that the transition matrix $G = cS + (1-c)\mathbf{1}\boldsymbol{z}^T$ will now depend on the blacklist of the walker. Let

$$G(\mathcal{B}) := cS(\mathcal{B}) + (1-c)\mathbf{1}\boldsymbol{z}^T \tag{6.1}$$

be the transition matrix corresponding to $\mathcal{B}$, we have for every $i, j \in \mathcal{N}$

$$S(\mathcal{B})_{ij} = \begin{cases} S_{ij} + \sum_{k \in \mathcal{B}} S_{ik} \cdot \boldsymbol{z}_j & \text{if } j \notin \mathcal{B}, \\ \sum_{k \in \mathcal{B}} S_{ik} \cdot \boldsymbol{z}_j & \text{if } j \in \mathcal{B}, \end{cases}$$

where $\sum_{k \in \mathcal{B}} S_{ik}$ is the probability to visit a forbidden node and to jump with an empty blacklist. Therefore, the probability to be in j at time $t+1$ given that the previous state was $Y_t = (i, \mathcal{B})$ is given by

$$\text{Prob}(i_{t+1} = j | Y_t = (i, \mathcal{B})) = G(\mathcal{B})_{ij}.$$

In order to determine the transition to the complete state Y_{t+1}, we also need the transition of the blacklist. Clearly, Y_{t+1} only depends on the previous state Y_t and therefore a trust walk is a particular Markov chain that we label $\mathcal{M}$ (see Section 2.4). The transitions are completely described in the following remark.

Remark. Four cases exist for the transition from $Y_t = (i, \mathcal{B})$ to $Y_{t+1} = (j, \mathcal{B}_{\text{new}})$ where $j \notin \mathcal{B}$:

1. If the new blacklist $\mathcal{B}_{\text{new}}$ contains the blacklist of j union $\mathcal{B}$ and some nodes of $\mathcal{B}$ are not in the blacklist of j, that is, $\mathcal{B}_{\text{new}} = \mathcal{B} \cup \mathcal{D}_j \neq \mathcal{D}_j$, then the walker moved without zapping or jumping in order to keep $\mathcal{B}$.

2. If $\mathcal{B}_{\text{new}}$ is exactly the blacklist of j and all the nodes in $\mathcal{B}$ are already in the blacklist of j, that is, $\mathcal{B}_{\text{new}} = \mathcal{B} \cup \mathcal{D}_j = \mathcal{D}_j$, then the walker has followed a link, jump or zap (every move is possible).

3. If $\mathcal{B}_{\text{new}}$ is exactly the blacklist of j, but some nodes in $\mathcal{B}$ are not in the blacklist of j, that is, $\mathcal{B}_{\text{new}} = \mathcal{D}_j \neq \mathcal{B} \cup \mathcal{D}_j$, then the walker has emptied $\mathcal{B}$ before jumping or zapping.

4. Else $\mathcal{B}_{\text{new}}$ is not compatible, that is, $\mathcal{B}_{\text{new}} \neq \mathcal{B} \cup \mathcal{D}_j$ and $\mathcal{B}_{\text{new}} \neq \mathcal{D}_j$.

According to these four cases, the transition probability

$$\text{Prob}(Y_{t+1} = (j, \mathcal{B}_{\text{new}}) | Y_t = (i, \mathcal{B}))$$

for the Markov chain $\mathcal{M}$ is given by

$$\begin{array}{ll} cS_{ij} & \text{if } \mathcal{B}_{\text{new}} = \mathcal{B} \cup \mathcal{D}_j \neq \mathcal{D}_j, \\ G(\mathcal{B})_{ij} & \text{if } \mathcal{B}_{\text{new}} = \mathcal{B} \cup \mathcal{D}_j = \mathcal{D}_j, \\ G(\mathcal{B})_{ij} - cS_{ij} & \text{if } \mathcal{B}_{\text{new}} = \mathcal{D}_j \neq \mathcal{B} \cup \mathcal{D}_j, \\ 0 & \hfill \text{otherwise,} \end{array}$$

where we use the matrix $G(\mathcal{B})$ defined in Eq. (6.1). Now if $j \in \mathcal{B}$, it implies that the trust walker has emptied his blacklist to reach j, therefore he zaps or he jumps and B_{new} must be equal to $\mathcal{D}_j$. In that case,

$$\text{Prob}(Y_{t+1} = (j, \mathcal{D}_j) | Y_t = (i, \mathcal{B})) = G(\mathcal{B})_{ij} - cS_{ij}.$$

See also the pseudo code in Table 6.1 that simulates realizations of the Markov chain $\mathcal{M}$.

No zapping with a probability c	
$\quad i_{new} = \text{rand}(i_t^{\text{th}} \text{ row of } S)$	*the walker follows an outlink*
$\quad$ if $i_{new} \notin \mathcal{B}_t$	*the node i_{new} is not forbidden*
$\qquad i_{t+1} = i_{new}$	
$\qquad \mathcal{B}_{t+1} = \mathcal{B}_t \cup \mathcal{D}_{i_{t+1}}$	*he updates the blacklist*
$\quad$ else	*otherwise he jumps*
$\qquad i_{t+1} = \text{rand}(\boldsymbol{z})$	
$\qquad \mathcal{B}_{t+1} = \emptyset$	*he empties the blacklist*
$\quad$ end	
Zapping with a probability $1 - c$	
$\quad \mathcal{B}_{t+1} = \emptyset$	*he empties the blacklist*
$\quad i_{t+1} = \text{rand}(\boldsymbol{z})$	*the walker zaps*

Table 6.1: The trust walk described as a Markov chain : one step from $Y_t = (i_t, \mathcal{B}_t)$ to $Y_{t+1} = (i_{t+1}, \mathcal{B}_{t+1})$. The function rand($\boldsymbol{v}$) returns a node in $\mathcal{N}$ according to the stochastic vector $\boldsymbol{v}$.

Steady state probabilities. We now show that a trust walk with zapping, that is $c \in [0, 1[$, leads to a stationary distribution that can be used to rank the nodes of the network. But the complexity of the calculation of that distribution can be combinatorial in the number of possible forbidden nodes in a blacklist. For that reason, we will see in the next section how the updates of the blacklist can be simplified in order to make the method usable for large networks.

We have seen that a trust walk is equivalent to the Markov chain $\mathcal{M}$. We will show that it has a unique stationary distribution when $c \in [0, 1[$. For that purpose, let us introduce the set $\mathcal{Z}$ that contains the reachable states after a zapping, i.e.,

$$\mathcal{Z} = \{(i, \emptyset) : \boldsymbol{z}_i > 0\},$$

and the following lemma that claims that the Markov chain has a unique final class and is aperiodic.

Lemma 6. *The Markov chain $\mathcal{M}$ with $c \in [0, 1[$ has a unique final class and is aperiodic.*

Proof: By definition, the nodes of a final class have no access to the nodes of other classes. But, from any state $y \in \mathcal{N} \times \mathcal{P}(\mathcal{N})$, there is a positive probability (because $c < 1$) to reach a state in $\mathcal{Z}$, it means that every state has access to $\mathcal{Z}$. Therefore, there is one final class $\mathcal{F}$ that is the set of states accessible from $\mathcal{Z}$:

$$\mathcal{F} = \{y \in \mathcal{N} \times \mathcal{P}(\mathcal{N}) : \text{Prob}(Y_t = y | Y_0 \in \mathcal{Z}) > 0, t \geq 0\}.$$

The aperiodicity of $\mathcal{M}$ directly follows from the presence of the zapping. ∎

Now, by Proposition 1, we know that the Markov chain $\mathcal{M}$ has a unique stationary distribution that gives us a probability of presence for every state $y \in \mathcal{N} \times \mathcal{P}(\mathcal{N})$ after an infinite time. Let

$$\boldsymbol{\pi}(\mathcal{B})_i := \lim_{t \to \infty} \text{Prob}(Y_t = (i, \mathcal{B}))$$

be the probability corresponding to the state $y = (i, \mathcal{B})$. Hence the vector $\boldsymbol{\pi} \in \mathbb{R}^n$ defined by

$$\boldsymbol{\pi}_i := \sum_{\mathcal{B}} \boldsymbol{\pi}(\mathcal{B})_i$$

for $i \in \mathcal{N}$, is the stationary distribution over the network without considering the blacklists. That vector is labelled the PageTrust vector and gives the probabilities of presence of a trust walker after an infinite time:

$$\boldsymbol{\pi}_i = \lim_{t \to \infty} \text{Prob}(i_t = i).$$

Since, that Markovian process restarts with probability $(1 - c)$, we can use Eq. (2.8) in Section 2.4 to calculate the stationary distribution vector:

$$\boldsymbol{\pi} = (1 - c)\boldsymbol{z} + (1 - c) \sum_{t=1}^{\infty} c^t \boldsymbol{x}^t, \tag{6.2}$$

where the vector $\boldsymbol{x}^t$ with $t \geq 1$ is the probability distribution of a trust walker who never zaps ($c = 1$), and the initial probability distribution of that walker is $\boldsymbol{z}$ and his initial blacklist is empty.

For the sequel, it will be convenient to define the Markov chain $\bar{\mathcal{M}}$ that corresponds to the Markov chain $\mathcal{M}$ without zapping, and with the initial state $Y_0 = (i_0, \emptyset)$ where i_0 is chosen in $\mathcal{N}$ according to $\boldsymbol{z}$. Let

$$\bar{Y}_t = (\bar{i}_t, \bar{\mathcal{B}}_t) \in \mathcal{N} \times \mathcal{P}(\mathcal{N})$$

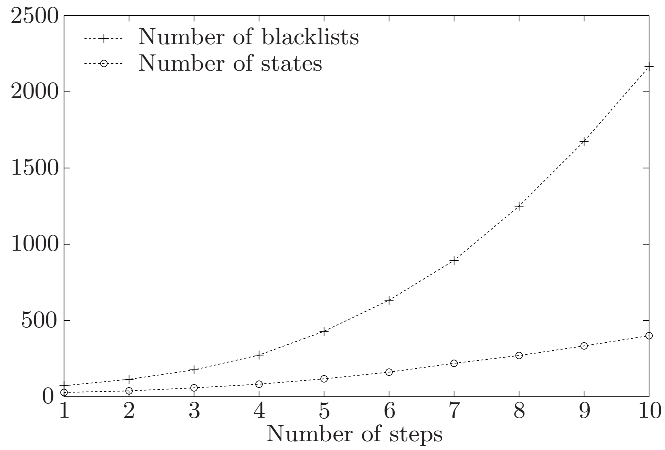


Figure 6.2: The number of states and blacklists that can be reached after t in $\mathcal{M}'$, i.e., $|\{y \in \mathcal{N} \times \mathcal{P}(\mathcal{N}) : \text{Prob}(\bar{Y}_t = y) > 0\}|$ and $|\{\mathcal{B} \in \mathcal{P}(\mathcal{N}) : \text{Prob}(\bar{\mathcal{B}}_t = \mathcal{B}) > 0\}|$, in a random network with 50 nodes, 100 positive links and 20 negative links.

be the random variables representing the state at time t for the Markov chain $\bar{\mathcal{M}}$. Hence, the i^{th} entry of the vector $\boldsymbol{x}^t$ in Eq. (6.2) corresponds to

$$\boldsymbol{x}_i^t = \text{Prob}(\bar{i}_t = i). \tag{6.3}$$

The PageTrust vector in Eq. (6.2) depends on the zapping vector $\boldsymbol{z}$ that has an increasing impact with $1 - c$, see example 11. As expected, the calculation of the vector $\boldsymbol{\pi}$ exponentially increases with the number of possible blacklists, see Fig. 6.2.

Example 11. The 4 nodes network in Fig. 6.3(ℓ) will be visited by a trust walker in the following way: once he visits node 4 at time t, his blacklist becomes $\mathcal{B}_t = \{1\}$. Hence, he will move through the 3 remaining nodes that are not forbidden in the network given in Fig. 6.3(r) until he will zap or jump and return with an empty blacklist to the network given in Fig. 6.3(ℓ). Let us first consider the (transposed) PageRank vectors for

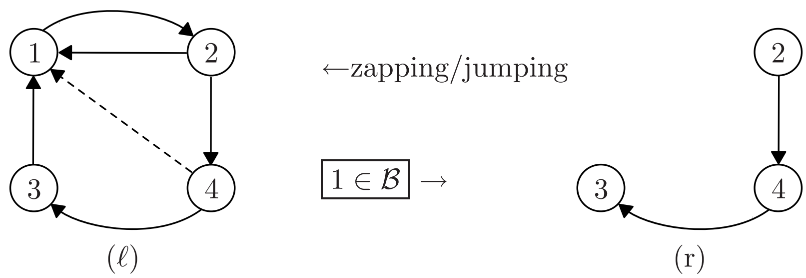


Figure 6.3: The 4 nodes network (ℓ) has a negative link from node 4 to node 1. Depending on his path, the trust walker will have at some time t the blacklist $\mathcal{B}_t = \{1\}$ and he will keep moving on the 3 nodes network (r). Then, he will return to (ℓ) by zapping or jumping.

the uniform zapping vector $\boldsymbol{z}_1$ and the zapping vector $\boldsymbol{z}_2 = \frac{1}{9}(1, 1, 1, 6)$:

	$c = .9$	$c = .5$
$\boldsymbol{z}_1$	$[\ .33\ \ .22\ \ .18\ \ .17\ ]$	$[\ .31\ \ .28\ \ .22\ \ .19\ ]$
$\boldsymbol{z}_2$	$[\ .32\ \ .29\ \ .19\ \ .20\ ]$	$[\ .22\ \ .16\ \ .24\ \ .37\ ]$

As expected, for $\boldsymbol{z}_1$, the order of importance follows the labelling of the nodes. However, when the zapping increases, then the differences between the ranks decrease. For $\boldsymbol{z}_2$, the zapping vector improves the position of node 4, especially when the zapping increases. Let us see now the (transposed) PageTrust vectors in Eq. (6.2) that take into account the negative link $(4, 1)$:

	$c = .9$	$c = .5$
$\boldsymbol{z}_1$	$[\ .26\ \ .30\ \ .24\ \ .20\ ]$	$[\ .28\ \ .28\ \ .24\ \ .21\ ]$
$\boldsymbol{z}_2$	$[\ .16\ \ .18\ \ .33\ \ .32\ ]$	$[\ .14\ \ .14\ \ .29\ \ .44\ ]$

Clearly, node 1 is penalized by its negative inlink. This increases when we consider $\boldsymbol{z}_2$. Moreover, the rank of node 2 also decreases because its reputation depends on the one of his unique parent, i.e., node 1, that is distrusted.

6.3 The simplified PageTrust

The goal of this section is to modify the trust walk defined above to calculate the steady state probabilities of a simplified trust walk in $O(n^2)$. The main difficulty, as observed in Fig. 6.2, comes from the combinatorial number of possible blacklists in the number of forbidden nodes. Therefore rather than consider all the possible cases, we will restrict ourselves to an incomplete description of the blacklists.

Let us remind that the PageTrust vector $\boldsymbol{\pi}$ with zapping in Eq. (6.2) requires the calculation of the stochastic vector $\boldsymbol{x}^t$ in Eq. (6.3) for a certain number of $t \geq 1$ according to the desired precision on $\boldsymbol{\pi}$. As seen in Eq. (6.3), the entry $\boldsymbol{x}_i^t$ gives the probability to be in node i at time t when we consider the Markov chain $\bar{\mathcal{M}}$. In order to determine $\boldsymbol{x}^{t+1}$, it is not sufficient to have $\boldsymbol{x}^t$, we need to compute for every possible state $(i, \mathcal{B})$ the probabilities

$$\boldsymbol{x}(\mathcal{B})_i^t := \operatorname{Prob}(\bar{Y}_t = (i, \mathcal{B})), \tag{6.4}$$

where $\bar{Y}_t = (\bar{i}_t, \bar{\mathcal{B}}_t)$ is the state at time t of the Markov chain $\bar{\mathcal{M}}$. Since the number of states can exponentially increases, rather than computing $\boldsymbol{x}(\mathcal{B})_i^t$, we will consider

$$\boldsymbol{x}(k)_i^t := \operatorname{Prob}(\bar{i}_t = i, k \in \bar{\mathcal{B}}_t), \tag{6.5}$$

for all $i, k \in \mathcal{N}$. As said before, we then only have an incomplete description of the blacklists that have been somehow aggregated:

$$\boldsymbol{x}(k)^t = \sum_{\mathcal{B}: k \in \mathcal{B}} \boldsymbol{x}(\mathcal{B})^t,$$

for all $k \in \mathcal{N}$.

The next subsection presents how $\boldsymbol{x}^{t+1}$ and $\boldsymbol{x}(k)^{t+1}$ can be estimated from $\boldsymbol{x}^t$ and $\boldsymbol{x}(k)^t$ for all $k \in \mathcal{N}$.

Required simplifications. We introduce the simplified iterations on $\boldsymbol{x}^t$ and $\boldsymbol{x}(k)^t$ for $k \in \mathcal{N}$ that will be given in Eq. (6.8,6.9). The first part develops the updating of $\boldsymbol{x}^t$ and the second part, the updating of $\boldsymbol{x}(k)^t$ for $k \in \mathcal{N}$. In the sequel, we will refer to a reasonable assumption that will simplify the equations and that we present here:

Assumption B. The events $k \in \bar{\mathcal{B}}_t$ and $i \notin \bar{\mathcal{B}}_t$ are independent for all $i, k \in \mathcal{N}$, $i \neq k$ and $t \geq 1$.

That assumption of statistical independence is commonly used in information retrieval (e.g., the naive Bayes classifier [66]), or in data fusion (e.g. the simple probabilistic model [101]), and it is considered for the same reason than here: simplifying the calculation. Under Assumption **B**, we remark that for any $i, j \in \mathcal{N}$, $i \neq k$ and $t \geq 1$, we have that $\text{Prob}(\bar{i}_t = j, k \in \bar{\mathcal{B}}_t, i \notin \bar{\mathcal{B}}_t)$

$$
\begin{aligned}
&= \text{Prob}(\bar{i}_t = j, k \in \bar{\mathcal{B}}_t) \cdot \text{Prob}(i \notin \bar{\mathcal{B}}_t | \bar{i}_t = j, k \in \bar{\mathcal{B}}_t) \\
&= \boldsymbol{x}(k)_j^t \cdot \text{Prob}(i \notin \bar{\mathcal{B}}_t | \bar{i}_t = j, k \in \bar{\mathcal{B}}_t) \quad \big(\text{by Eq. (6.5)}\big) \\
&= \boldsymbol{x}(k)_j^t \cdot \text{Prob}(i \notin \bar{\mathcal{B}}_t | \bar{i}_t = j) \quad \big(\text{by Ass. } \mathbf{B}\big) \\
&= \boldsymbol{x}(k)_j^t \cdot (1 - \text{Prob}(i \in \bar{\mathcal{B}}_t | \bar{i}_t = j)) \\
&= \boldsymbol{x}(k)_j^t \cdot \left(1 - \frac{\text{Prob}(i \in \bar{\mathcal{B}}_t, \bar{i}_t = j)}{\text{Prob}(\bar{i}_t = j)}\right) \\
&= \boldsymbol{x}(k)_j^t \cdot \left(1 - \frac{\boldsymbol{x}(i)_j^t}{\boldsymbol{x}_j^t}\right) \quad \big(\text{by Eq. (6.3) and Eq. (6.5)}\big) \\
&= \boldsymbol{x}(k)_j^t \, \boldsymbol{\lambda}(i)_j^t,
\end{aligned} \tag{6.6}
$$

where we defined

$$
\boldsymbol{\lambda}(i)_j^t := 1 - \frac{\boldsymbol{x}(i)_j^t}{\boldsymbol{x}_j^t} = \text{Prob}(i \notin \bar{\mathcal{B}}_t | \bar{i}_t = j) \tag{6.7}
$$

that is the probability to trust $i \in \mathcal{N}$ at time t ($i \notin \bar{\mathcal{B}}_t$) when the walker is in $j \in \mathcal{N}$ ($\bar{i}_t = j$).

In order to present the updating of the vector $\boldsymbol{x}^t$ to $\boldsymbol{x}^{t+1}$, we decompose $\boldsymbol{x}^{t+1}$ as a sum of two contributions:

$$
\boldsymbol{x}^{t+1} = \bar{\boldsymbol{x}}^{t+1} + \hat{\boldsymbol{x}}^{t+1}.
$$

The entry $\bar{\boldsymbol{x}}_i^{t+1}$ is then the probability to visit $i \in \mathcal{N}$ by following a link $(j, i) \in \mathcal{L}^+$ with i not in the blacklist $\bar{\mathcal{B}}_t$. The second contribution $\hat{\boldsymbol{x}}_i^{t+1}$

is the probability to jump in $i \in \mathcal{N}$. Therefore, we have for all $i \in \mathcal{N}$

$$\begin{aligned}
\bar{\boldsymbol{x}}_i^{t+1} &= \text{Prob}(\bar{i}_{t+1} = i, \text{ following a link}) \\
&= \sum_{j \to i} \text{Prob}(\bar{i}_{t+1} = i, \bar{i}_t = j, \text{ following a link}) \\
&= \sum_{j \to i} \text{Prob}(\bar{i}_t = j, i \notin \bar{\mathcal{B}}_t) \cdot \text{Prob}(\bar{i}_{t+1} = i | \bar{i}_t = j, \text{ following a link}) \\
&= \sum_{j \to i} \text{Prob}(\bar{i}_t = j, i \notin \bar{\mathcal{B}}_t) \cdot S_{ji} \\
&= \sum_{j \to i} \text{Prob}(\bar{i}_t = j) \cdot \text{Prob}(i \notin \bar{\mathcal{B}}_t | \bar{i}_t = j) \cdot S_{ji} \\
&= \sum_{j \to i} \boldsymbol{x}_j^t \, \boldsymbol{\lambda}(i)_j^t \, S_{ji} \quad \big(\text{by Eq. (6.5) and Eq. (6.7)}\big),
\end{aligned}$$

and

$$\begin{aligned}
\hat{\boldsymbol{x}}_i^{t+1} &= \text{Prob}(\bar{i}_{t+1} = i, \text{ jumping}) \\
&= \text{Prob}(\bar{i}_{t+1} = i | \text{jumping}) \cdot \text{Prob}(\text{jumping}) \\
&= \boldsymbol{z}_i \cdot \sum_{k: j \to k} \text{Prob}(\bar{i}_t = j, k \in \bar{\mathcal{B}}_t) \, S_{jk} \\
&= \boldsymbol{z}_i \sum_{k: j \to k} \boldsymbol{x}(k)_j^t S_{jk} \quad \big(\text{by Eq. (6.5)}\big) \\
&= \boldsymbol{z}_i \gamma^{t+1}
\end{aligned}$$

where we defined $\gamma^{t+1} := \sum_{j \to k} \boldsymbol{x}(k)_j^t S_{jk}$ that is the probability of jumping at time $t+1$ and it does not depend on i.

Therefore, the sum of both contributions gives for all $i \in \mathcal{N}$:

$$\boldsymbol{x}_i^{t+1} = \sum_{j \to i} \boldsymbol{x}_j^t \, \boldsymbol{\lambda}(i)_j^t \, S_{ji} + \gamma^{t+1} \boldsymbol{z}_i. \tag{6.8}$$

Secondly, we describe the updating of the vector $\boldsymbol{x}(k)^t$ to $\boldsymbol{x}(k)^{t+1}$ for all $k \in \mathcal{N}$. In the same way, we consider two contributions such that, for all $k \in \mathcal{N}$,

$$\boldsymbol{x}(k)^{t+1} = \bar{\boldsymbol{x}}(k)^{t+1} + \hat{\boldsymbol{x}}(k)^{t+1}.$$

The entry $\bar{\boldsymbol{x}}(k)_i^{t+1}$ is the probability to visit $i \in \mathcal{N}$ by following a link $(j, i) \in \mathcal{L}^+$ with i not in the blacklist $\bar{\mathcal{B}}_t$, and $k \in \bar{\mathcal{B}}_{t+1}$. Then, the entry

$\hat{\boldsymbol{x}}(k)_i^{t+1}$ is the probability to jump in $i \in \mathcal{N}$ from a forbidden node and to have $k \in \bar{\mathcal{B}}_{t+1}$.

Let us first remark that when i distrusts k, i.e., $(i,k) \in \mathcal{L}^-$, then the probability to be in i at time $t+1$ or the probability to be in i and to distrust k at time $t+1$ are the same, therefore

$$\boldsymbol{x}(k)_i^{t+1} = \boldsymbol{x}_i^{t+1} \qquad \text{if } (i,k) \in \mathcal{L}^-.$$

Otherwise, the only manner to obtain $k \in \bar{\mathcal{B}}_{t+1}$ is to already have $k \in \bar{\mathcal{B}}_t$. The entry $\bar{\boldsymbol{x}}(k)_i^{t+1}$ is then the probability to visit i by following a link $(j,i) \in \mathcal{L}^+$ with i not in the blacklist $\bar{\mathcal{B}}_t$, but well $k \in \bar{\mathcal{B}}_t$. Therefore, by using Assumption **B** and Eq. (6.6), we have

$$\begin{aligned} \bar{\boldsymbol{x}}(k)_i^{t+1} &= \sum_{j \to i} \mathrm{Prob}(\bar{i}_t = j, k \in \bar{\mathcal{B}}_t, i \notin \bar{\mathcal{B}}_t)\, S_{ji} \\ &= \sum_{j \to i} \boldsymbol{x}(k)_j^t\, \boldsymbol{\lambda}(i)_j^t\, S_{ji}. \end{aligned}$$

for all $i,k \in \mathcal{N}$, $i \neq k$. Then $\hat{\boldsymbol{x}}(k)_i^{t+1} = 0$ because we assume that the blacklist is emptied before jumping and therefore $k \notin \bar{\mathcal{B}}_{t+1} = \emptyset$.

Finally, we have for all $i,k \in \mathcal{N}$, $i \neq k$:

$$\boldsymbol{x}(k)_i^{t+1} = \begin{cases} \boldsymbol{x}_i^{t+1} & \text{if } (i,k) \in \mathcal{L}^-, \\ \sum_{j \to i} \boldsymbol{x}(k)_j^t\, \boldsymbol{\lambda}(i)_j^t\, S_{ji} & \text{else.} \end{cases} \tag{6.9}$$

The s-PageTrust vector and its interpretation. Starting with the zapping vector, that is, $\boldsymbol{x}^0 = \boldsymbol{z}$, we have $\boldsymbol{x}(k)_i^0 = \boldsymbol{x}_i^0$ for all $(i,k) \in \mathcal{L}^-$ else $\boldsymbol{x}(k)_i^0 = 0$. The sequences $(\boldsymbol{x}^t)$ and $(\boldsymbol{x}(k)^t)$ are given by the iteration steps defined by Eq. (6.8,6.9). We will reformulate these iteration steps by defining the time-varying matrix

$$S_t = S \circ \left[\boldsymbol{\lambda}(1)^t \cdots \boldsymbol{\lambda}(n)^t\right], \tag{6.10}$$

where the entries (i,j) of S_t, i.e., $\boldsymbol{\lambda}(j)_i^t\, S_{ij}$, represent the transition by following a link at time t. Hence, we have

$$\left[\boldsymbol{x}^{t+1}\right]^T = \left[\boldsymbol{x}^t\right]^T S_t + \gamma^{t+1} \boldsymbol{z}^T, \tag{6.11}$$

$$\boldsymbol{x}(k)_i^{t+1} = \begin{cases} \boldsymbol{x}_i^{t+1} & \text{if } (i,k) \in \mathcal{L}^-, \\ \left(\left[\boldsymbol{x}(k)^t\right]^T S_t\right)_i & \text{else,} \end{cases} \tag{6.12}$$

where γ^{t+1} can be equivalently calculated by taking

$$1 - \left[\boldsymbol{x}^t\right]^T S_t \mathbf{1}.$$

Hence, the s-PageTrust vector is given by

$$\boldsymbol{\pi} = (1-c)\boldsymbol{z} + (1-c)\sum_{t=1}^{\infty} c^t \boldsymbol{x}^t, \tag{6.13}$$

that differs from the PageTrust vector in Eq. (6.2) by the definition of the vectors $\boldsymbol{x}^t$ that is now an estimate of the one defined in Eq. (6.3). However the vectors $\boldsymbol{x}^t$ given by the iterations in Eq. (6.11,6.12) can still be interpreted as the probability distribution of some modified trust walk that we describe here.

Our walker moves in the network $\mathcal{G}$ simultaneously with an infinite number of other walkers who behave identically. When he visits a node $j \in \mathcal{N}$, he updates his blacklist by adding the distrusted nodes of j. Let $\mathcal{W}$ be the set of all the walkers in j at that time. Then, before leaving j and reaching a new node, say i – chosen according to the transition matrix S –, he decides whether or not i is forbidden with a probability p. That probability is given by the proportion of walkers trusting i in $\mathcal{W}$ (unlike before where that choice only depended on his own blacklist). Hence, he visits i with a probability p, or he jumps (with a new blacklist) according to $\boldsymbol{z}$ with a probability $1-p$.

Let us finally remark that the s-PageTrust vector $\boldsymbol{\pi}$ in Eq. (6.13) is the probability distribution vector of the same walker just described before but with the following additional rule: at every step, all the walkers zap with a probability $1-c$.

Example 12. In the case of the 4 nodes network in Fig. 6.3(ℓ), the (transposed) s-PageTrust vectors calculated from Eq. (6.13) will give the same results as in Example 11. This is because Assumption **B** is not necessary and therefore the vectors $\boldsymbol{x}^t$ in Eq. (6.3) and (6.11) are equivalent.

Degree of conviction. We have seen in our simplified trust walk that our walker, being at time t in j and moving from j to i, decides with a probability p to trust i. The scalar p is given by the proportion of walkers in j who trust i. Therefore, for an infinite number of walkers, we can use Eq. (6.7) and we have

$$p = \boldsymbol{\lambda}(i)^t_j.$$

We will suppose now that when a walker distrusts some node, he has some *degree of conviction* about his blacklist. That degree of conviction will change the probability p so that when it is high, a small proportion of walkers is enough to convince many walkers to not visit i. On the contrary, if the degree of conviction is relatively small, then the proportion of distrusting walkers does not matter much.

Let $\beta \in \mathbb{R}_{\geq 0}$ represents the degree of conviction, then we pose

$$p = \left(\boldsymbol{\lambda}(i)_j^t\right)^\beta,$$

and we have, for relatively small values of $\boldsymbol{\lambda}(i)_j^t$ (which is often the case in large and sparse networks), that

$$p \approx 1 - \beta \frac{\boldsymbol{x}(i)_j^t}{\boldsymbol{x}_j^t}.$$

This means that approximately β times the proportion of walkers who distrust i will not visit i. The parameter β is taken into account for the s-PageTrust vector in Eq. (6.13) by merely changing in Eq. (6.11,6.12) the time-varying transition matrix S_t. That matrix is now defined by replacing the terms $\boldsymbol{\lambda}(k)_i^t$ by

$$\left(\boldsymbol{\lambda}(k)_i^t\right)^\beta$$

for all $i, k \in \mathcal{N}$. We remark then that for $\beta = 1$, we recover the s-PageTrust defined previously, and for $\beta = 0$, we recover the PageRank where negative links are merely ignored.

Example 13. Let us again consider the 4 nodes network in Fig. 6.3(m). The next table shows the (transposed) ranking vectors calculated from Eq. (6.13) with a degree of conviction $\beta = 0.1$ for the uniform zapping vector $\boldsymbol{z}_1$ and the zapping vector $\boldsymbol{z}_2 = \frac{1}{9}(1, 1, 1, 6)$:

	$c = .9$	$c = .5$
$\boldsymbol{z}_1$	$\begin{bmatrix} .31 & .31 & .20 & .18 \end{bmatrix}$	$\begin{bmatrix} .29 & .28 & .24 & .20 \end{bmatrix}$
$\boldsymbol{z}_2$	$\begin{bmatrix} .25 & .25 & .25 & .25 \end{bmatrix}$	$\begin{bmatrix} .15 & .14 & .28 & .43 \end{bmatrix}$

The mutual negative link between node 1 and 4 penalizes them less than the PageTrust. The results are between the PageRank and the PageTrust given in example 11.

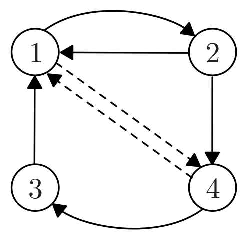


Figure 6.4: The 4 nodes network has a mutual negative link between nodes 1 and 4. Depending on his path, the trust walker will have at some time t the blacklist $\bar{\mathcal{B}}_t = \{4\}$ or $\bar{\mathcal{B}}_t = \{1\}$.

6.4 Properties and examples

We discuss in this section several properties of the s-PageTrust defined in Eq. (6.13). These properties will be often illustrated by small examples in which we compare several ranking vectors. We look at (1) the difference with the initial PageTrust without simplifications; (2) the convergence and the complexity; (3) the impact of the zapping and the degree of conviction; (4) the robustness when users manipulate the ranking vector by their outlinks; (5) a large dataset of the website Epinions.

In the sequel, the rank of a node, say i, will be the i^{th} entry of the s-PageTrust vector in Eq. (6.13).

Effect of the simplifications. As seen before, the simplifications on the iterations in Eq. (6.8,6.9) are based on one important assumption concerning the independency of two events, that is, Assumption **B** that can be reformulated as

$$\text{Prob}(i, k \in \bar{\mathcal{B}}_t) = \text{Prob}(i \in \bar{\mathcal{B}}_t) \cdot \text{Prob}(k \in \bar{\mathcal{B}}_t)$$

for all $i, k \in \mathcal{N}$, $i \neq k$ and $t \geq 1$. This is not satisfied, for example, when there is a mutual negative link as in Fig. 6.4. In that case, we have $\text{Prob}(1, 4 \in \bar{\mathcal{B}}_t) = 0$ while $\text{Prob}(1 \in \bar{\mathcal{B}}_t) \cdot \text{Prob}(4 \in \bar{\mathcal{B}}_t) \neq 0$.

In order to measure that difference, we calculate both ranking vectors on 1000 realizations of random directed networks with 50 nodes, 150 links and 5 mutual negative links. These negative links lead to a maximum of 3^5 combinations of possible blacklists. The $Y-$axis of Fig. 6.5 gives the quartiles of the distribution of the s-PageTrusts over the PageTrusts for all nodes. In most cases, that ratio belongs to [0.9 1.1] when the zapping

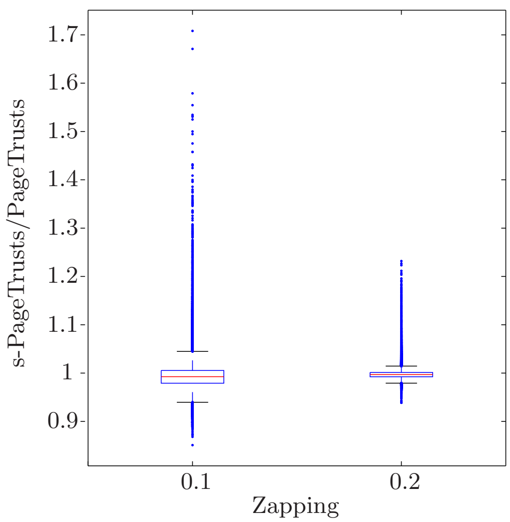


Figure 6.5: Quartiles over 1000 realizations of random directed networks with 50 nodes, 150 links and 5 mutual negative links. The whiskers have a maximum length of 1.5 times the interquartile range. The ration (s-PageTrust/PageTrust) for all nodes with the probability of zapping equals to 0.1 and 0.2.

is 0.1 and this interval becomes even smaller when the zapping increases to 0.2. If we order the nodes according to their PageTrusts, then we observe in Fig. 6.6 that the interquartile range is smaller for nodes with high PageTrusts compared to other nodes. Therefore, the estimation is better for the top of the ranking list that for the last nodes.

Convergence and complexity. The calculation of the s-PageTrust vector in Eq. (6.13) is based on the vector $\boldsymbol{x}^t$ that is iteratively computed by Eq. (6.11,6.12). For a sparse network $\mathcal{G} = (\mathcal{N}, \mathcal{L})$, i.e., $m << n^2$, we will see that one iteration step requires $O(n^2)$ operations. More exactly, if $\mathcal{N}_d$ denotes the set of nodes with at least one negative inlink in $\mathcal{G}$ and $n_d = |\mathcal{N}_d|$, then that complexity becomes $O(n_d \cdot n)$.

For one iteration step, we first need to compute the time-varying transition matrix S_t. This is done by calculating the n_d vectors $\boldsymbol{\lambda}(k)^t$ in Eq. (6.7), that requires $n_d \cdot O(n)$ operations. Then, the matrix componentwise product in Eq. (6.10) requires $O(n)$ operations because S is sparse. The updating of $\boldsymbol{x}^{t+1}$ is the product of a vector with S_t that is

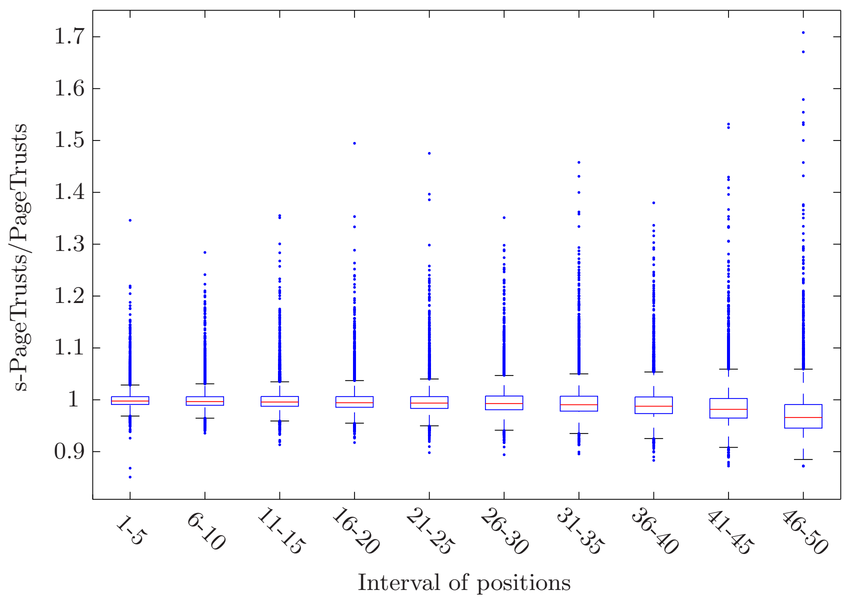


Figure 6.6: Quartiles over 1000 realizations of random directed networks with 50 nodes, 150 links and 5 mutual negative links. The whiskers have a maximum length of 1.5 times the interquartile range. The ration (s-PageTrust/PageTrust) for all nodes with sets of 5 nodes ordered by their PageTrusts with the probability of zapping equals to 0.1.

sparse by construction and the addition of another vector. Therefore, the number of operations is $O(n)$. We can show with a similar argument that the updating of $\boldsymbol{x}(k)^{t+1}$ requires $O(n)$ operations. Finally, the total number of operation for one iteration step is $O(n_d \cdot n)$ and the expensive step is the calculation of the n_d vectors $\boldsymbol{\lambda}(k)^t$.

After s iteration steps, we obtain the probability distribution vectors $\boldsymbol{x}^1, \cdots, \boldsymbol{x}^s$ and the truncated sum

$$\tilde{\boldsymbol{\pi}} = (1-c)\boldsymbol{z} + (1-c)\sum_{t=1}^{s} c^t \boldsymbol{x}^t.$$

The vector $\tilde{\boldsymbol{\pi}}$ estimates the s-PageTrust vector $\boldsymbol{\pi}$ in Eq. (6.13) with $\|\boldsymbol{\pi} - \tilde{\boldsymbol{\pi}}\|_1 = \frac{c^{s+1}}{1-c}$. This shows that the rate of convergence is q-linear in the number of iteration steps defined in Eq. (6.11,6.12) (see Section 2.5).

Zapping and degree of conviction. The s-PageTrust algorithm depends on the zapping α and the degree of conviction β. Both parameters allow to control the effect of negative links. The parameter α ranges from 0 (where there is no zapping) to 1 (where the walker zaps at each step). The addition of zapping increases the probability of emptying the blacklist and therefore it decreases the effect of negative links. The parameter β ranges from 0 to ∞, from the recovering of the PageRank algorithm to the absolute contagion where one random walker is enough to convince all the other ones. Obviously, the higher β, the more penalizing the negative links. We illustrate in the next example the sensitivity of the s-PageTrust vector with respect to these two parameters.

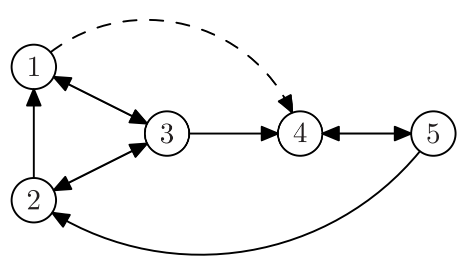


Figure 6.7: Network with one negative link from node 1 to node 4, see Example 14.

Example 14. The 5 nodes network in Fig. 6.7 leads to interesting results shown in Fig. 6.8: (a) uses $\beta = 0$ (equivalent to the PageRank), it shows

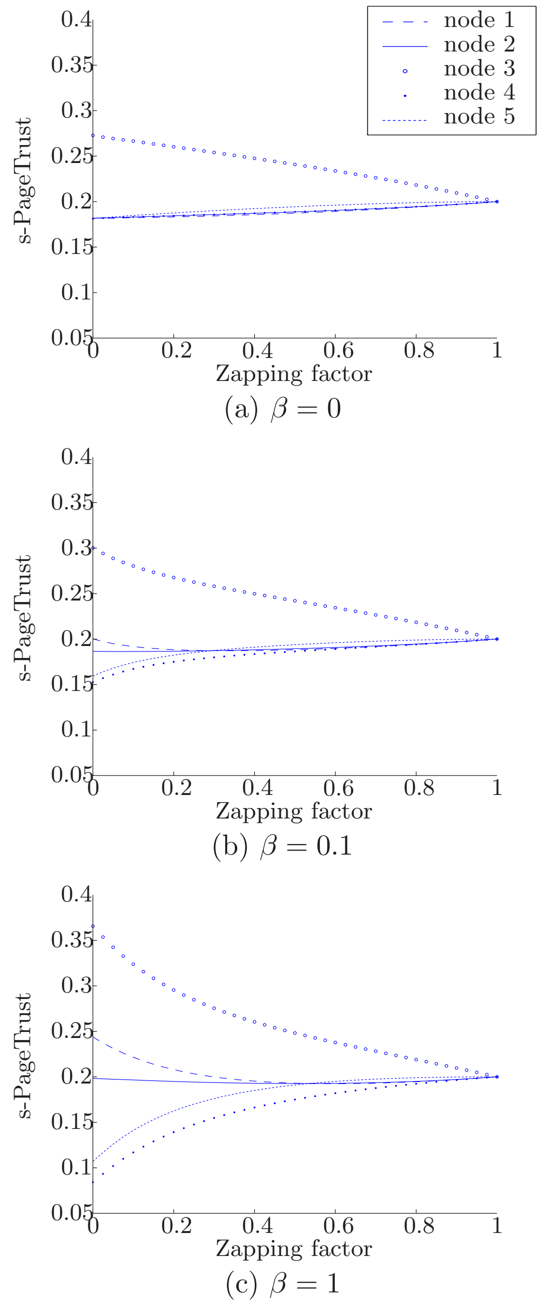
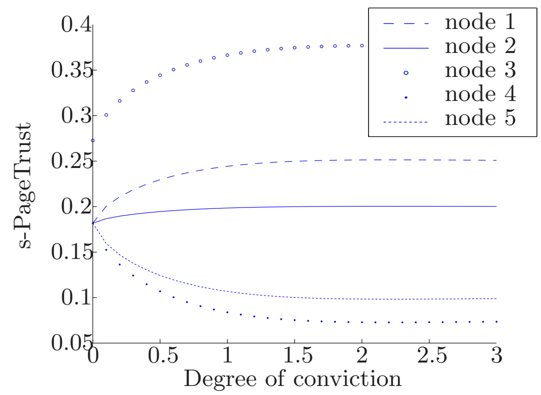
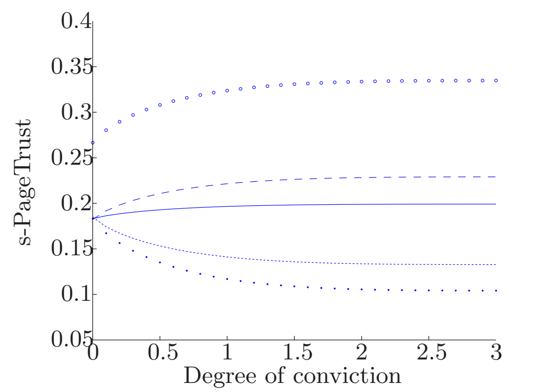


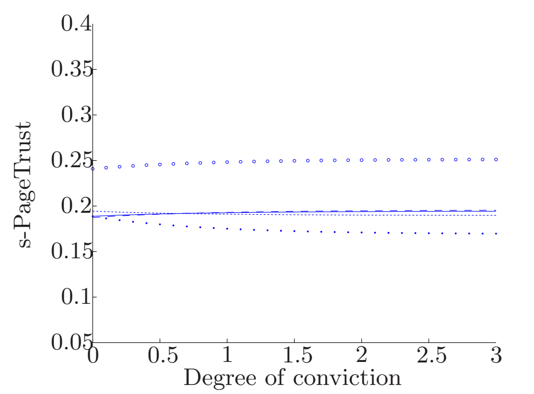
Figure 6.8: s-PageTrust for every node in the network of Fig. 6.7 with a continuous range of zapping factors and three degrees of conviction.



(a) $c = 1$



(b) $c = 0.9$



(c) $c = 0.5$

Figure 6.9: s-PageTrust for every node in the network of Fig. 6.7 with a continuous range of degrees of conviction and three zapping factors.

the ranks for all possible values of zapping. Node 3 is largely above the four other nodes that share almost the same rank. Naturally, zapping decreases the differences of PageTrust between the nodes until having them all equal. (b) introduces some degree of conviction ($\beta = 0.1$), and increases the effect of the negative link to the ranking. Hence, we better separate the ranks of nodes 1, 2, 4 and 5 for low values of zapping. As expected, node 4 is the most penalized, followed by its unique child, node 5. (c) increases the degree of conviction ($\beta = 1$) and the effect of the negative link. Therefore the differences of ranks become larger, even for relatively large value of zapping. Fig 6.9 shows the results when the degree of conviction varies (a) puts $c = 1$ (there is no zapping), then the effect of the negative link is maxima. The separation between the ranks becomes larger if β increases until reaching some stationary ranks. (b-c) The same phenomenon of separation occurs, however it is diminished by the addition of zapping.

Robustness to attacks. Ideally, as explained in the introduction, the rank must be robust to attacks. Two possible attacks with negative links are to increase its own rank or to decrease the rank of competing nodes. Linkage strategy with positive links have already been investigated [5, 46].

Since the rank of node i can be interpreted as the proportion of walkers in i once the steady state reached, one strategy could be to try to lure back walkers to itself via negative links. A similar idea was used in [5] with positive links. There, it is explained that optimal linkage strategy is obtained for a node when it points to one of its parents in order to make the random walker return to itself. With negative links, such a strategy seems less obvious. For instance, a natural idea consists in pointing negatively to nodes that represent a leak for node i, that is nodes that send the walkers far away from node i. But that strategy, illustrated for the network in Fig. 6.10, does not help since a walker that distrusts the leaking nodes (node 3 in the figure) will not choose between the remaining outlinks (link $(3, 2)$ in the figure) but it will rather jump.

Example 15. Let us consider the network illustrated in Fig. 6.10. The results are shown in Fig. 6.11. The rank lost by node 3 is earned by node 1, but also by nodes 5 and 6. We see that negative linkage does not necessarily increase its own rank, however it allows to decrease the rank of the nodes that are negatively pointed to. This becomes interesting

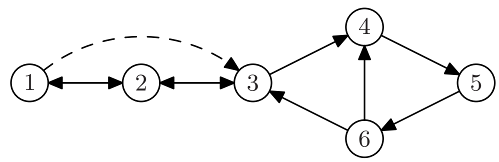
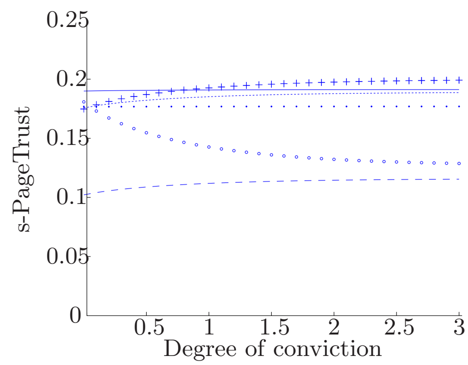


Figure 6.10: Node 3 is a leak for node 1 since it can send walkers to the nodes $4, 5, 6$ what makes the walkers move away from node 1, see Example 15

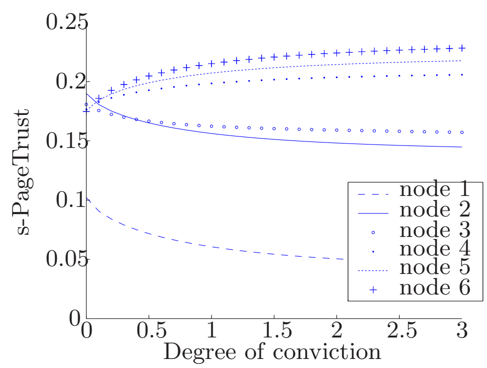
when they are compared in the same ranking list. A reasonable way to avoid such a behavior is to consider negative links in both directions. In other words, if node i declares itself against node k, we could consider that automatically node k will declare itself against node i. That is a mutual negative link or a mutual distrust. In Fig. 6.11(b), the mutual distrust between node 1 and 3 is not interesting for node 1 nor for node 3. Interestingly, node 2 that is between these two distrusted nodes is also penalized by the mutual negative link. The main benefit is actually made by nodes 4, 5 and 6.

Simulation. A dataset coming from the website Epinions is used to observe the influence of negative links on the ranking list. That website proposes to its members to review commercial items such as books or movies, but also to give a positive or negative opinion to another member. We subtract from the dataset about 38 000 members who are strongly connected by around 540 000 positive and negative votes.

We then compare the positions of each member when we use the s-PageTrust with $\beta = 10$, $c = 0.85$ and the PageRank, corresponding to $\beta = 0$. Fig. 6.12 shows that the more negative links, the more positions you may drop in average. For example, in the ranking list for $\beta = 10$, the users having one negative inlink drop in average by 248 positions in comparison to the ranking list for the PageRank. On the contrary, when we take into account the negative links, the users without negative inlink take advantage of the situation, see Fig. 6.12. We see that the nodes with negative inlinks decreases in average; however some nodes among them improve their position when $\beta = 10$. The reason of this improvement is twofold: the negative links pointing to a node i are given by nodes that receive much less visits of the trust walker that node i, and



(a) $\mathcal{L}^- = \{(1,3)\}$



(b) $\mathcal{L}^- = \{(1,3),(3,1)\}$

Figure 6.11: Ranks for a zapping of $\alpha = 0.1$ without (a) and with (b) mutual negative links.

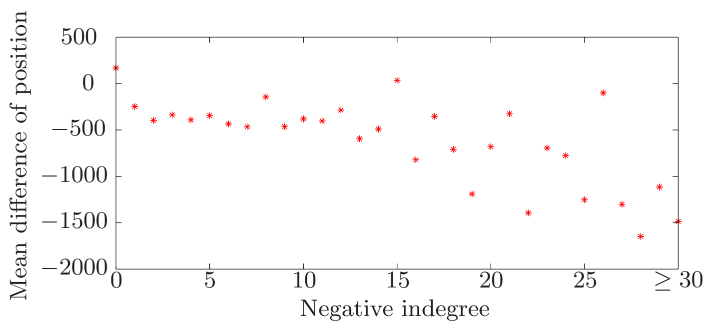


Figure 6.12: Average difference of positions by nodes having the same number of negative inlinks when we compare the simplified trust walk ($\beta = 10$, $c = 0.85$) with the PageRank ($\beta = 0$, $c = 0.85$).

these nodes are far away from node i. Therefore that distrusted node i avoids the effect of its negative inlinks while other penalized nodes drop back in the ranking list.

6.5 Extensions and variants

The PageTrust vector in Eq. (6.2) and the s-PageTrust vector in Eq. (6.13) are based on several choices and two parameters that were presented in the previous sections: the transition matrix S, the zapping vector $\boldsymbol{z}$, emptying or not the blacklist before zapping and before jumping, the zapping factor $1 - c$ and the degree of conviction β. In this section, we introduce some variants and extensions that will depend on the other possible choices that we did not explore.

The first subsection deals with the choice to keep the blacklist after zapping and jumping (instead of emptying it), and to have no zapping, i.e., $c = 1$. The second subsection uses the s-PageTrust vector as a local trust metric. The ranks reflect then the opinion of a particular node. Finally, the third subsection extends the PageTrust and s-PageTrust algorithms by replacing the transition matrix S of a trust walker by any nonnegative matrix B describing a trust flow through the network. This leads us to the FlowTrust and s-FlowTrust algorithms.

Emptying the blacklist and having no zapping. So far, we have considered that the PageTrust and the s-PageTrust vectors are the stationary distributions of walkers who empty their blacklists before zapping and jumping. However, the three other choices deserve some attention: (1) the blacklist is emptied before zapping, but not before jumping; (2) the blacklist is not emptied before zapping, but well before jumping; (3) the blacklist is never emptied. Proposition (1) requires minor changes in the PageTrust and the s-PageTrust algorithms. Moreover, the results are generally close, especially when the zapping factor increases making negligible the jumping part of the walk. On the other hand, propositions (2) and (3) largely modify the previous equations and do not guarantee any rate of convergence for the s-PageTrust. Moreover, in proposition (3), the blacklist is never emptied and since $|\mathcal{B}_t|$ remains bounded for $t \geq 0$, we have that the sequence $(\mathcal{B}_t)$ tends to some final blacklist $\mathcal{B}$, i.e., $\lim_{t\to\infty} \mathcal{B}_t = \mathcal{B}$. The probability $w_{\mathcal{B}} = \text{Prob}(\lim_{t\to\infty} \mathcal{B}_t = \mathcal{B})$ to reach that final blacklist will depend on the initial state. Once a final blacklist is reached, it does not change and we can associate a fixed transition matrix. If that matrix is irreducible for every final blacklist, the stationary distribution is unique. Let $\boldsymbol{\pi}(\mathcal{B})$ be that distribution for the final blacklist $\mathcal{B}$, then the PageTrust vector is given by

$$\boldsymbol{\pi} = \sum_{\mathcal{B}\subset\mathcal{N}} w_{\mathcal{B}}\, \boldsymbol{\pi}(\mathcal{B}). \tag{6.14}$$

We see that the PageTrust vector depends on $w_{\mathcal{B}}$ and hence, it also depends on the initial state, see Example 16.

Example 16. The 4 nodes network in Fig. 6.5(m) has a mutual negative link between nodes 1 and 4. Therefore, a trust walker, depending on his initial sate, will have a final blacklist $\mathcal{B} = \{4\}$ (ℓ) or $\mathcal{B} = \{1\}$ (r). Then he will jump (represented by smaller arrows) every time he is supposed to visit a forbidden node, that is node 4 in (ℓ) and node 1 in (r).

Local trust metric. Thus far, the previous examples provide a global trust metric. By global trust metric, we mean a measure of trust that does not depend on the point of view of any particular user. This is of interest when one has no a priori about the network, like for instance in the case of the World Wide Web where no webpages are trusted a priori. However, the zapping vector $\boldsymbol{z}$ allows us to distribute some initial trust over the nodes.

In contrast, a local trust metric depends on the opinion of a user in the network. A natural idea, proposed in [89] with the PageRank

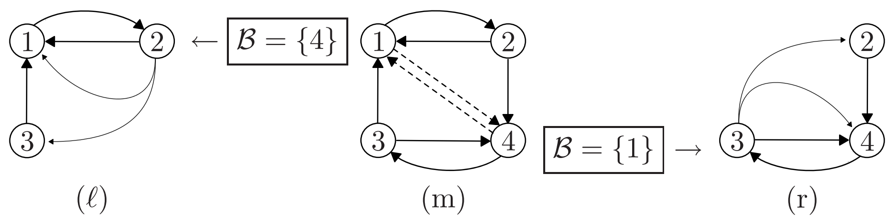


Figure 6.13: The 4 nodes network (m) has a mutual negative link between nodes 1 and 4. According to his blacklist, the trust walker will move to the network (ℓ) or (r), see Example 16.

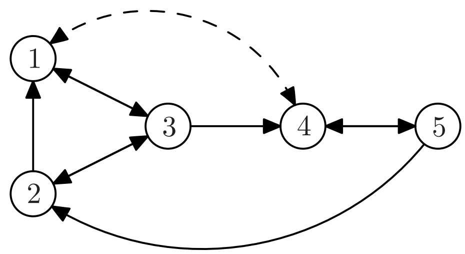


Figure 6.14: Network with one mutual distrust between node 1 and 4, see Example 17.

algorithm, is to put $\boldsymbol{z}_i$ equal to 1 and the rest of the entries of $\boldsymbol{z}$ equal to 0, i.e., $\boldsymbol{z}_i = \boldsymbol{e}_i$. In that way, the random walker starts in node i that is considered as the reference. Thereafter, there is a probability $1 - c$ that he comes back to node i after each iteration. In a trust walk, for example, no node distrusted by node i will be visited by the walker. Therefore the zapping allows us to favor or not *trust proximity*: if c is close to 0 the walker often comes back to the source node i, and if c is close to 1 he will more probably go further through some chains of trust, see Example 17.

Example 17. Table 6.2 illustrates two different opinions depending on the source node for the network in Fig. 6.14. These two opinions are calculated for $\beta = 1$ and are also compared with the method in [89] that corresponds to $\beta = 0$. We remark that node 2 favors node 1 rather than node 4 and therefore, the trust rank of node 5 also decreases when $\beta = 1$. On the contrary, the opinion of node 5 favors node 4 instead of node 1 which is not a direct child of node 5. Moreover, a switch of

	$\beta = 0$	$\beta = 1$
1	**.20**	**.23**
2	.24	.32
3	.29	.36
4	**.14**	**.04**
5	.13	.04

(ℓ) node 2

	$\beta = 0$	$\beta = 1$
1	**.15**	**.09**
2	.18	.21
3	.22	.17
4	**.18**	**.18**
5	.27	.35

(r) node 5

Table 6.2: Ranks for $c = 0.9$ depending on the opinions of node 2 and 5. This is calculated from the network in Fig. 6.14, see Example 17.

position occurs between node 2 and node 3 when $\beta = 1$.

The TrustFlow and s-TrustFlow algorithms. The transition matrix uses for the motion through a link of a trust walker was described by the matrix S with $S_{ij} = 1/\boldsymbol{d}_i$ if $(i, j) \in \mathcal{L}^+$ and $S_{ij} = 0$ else. Clearly, that matrix can be replaced by any other row stochastic matrix P to define another way to walk over the network (see Eq. (2.4) in Section 2.2).

More generally, we propose to replace S by a nonnegative matrix B, i.e., $B_{ij} \geq 0$ for all $i, j \in \mathcal{N}$. The interpretation in terms of walk through a network is not valid anymore, but we then commonly use a flow interpretation: the entry B_{ij} is the positive flow from i to j and the Perron vector (which is unique if B is irreducible)

$$\begin{aligned} \boldsymbol{\pi}^T &= \boldsymbol{\pi}^T B / \|\boldsymbol{\pi}^T B\|_1, \\ \boldsymbol{\pi}^T \mathbf{1} &= 1, \end{aligned} \tag{6.15}$$

represent the flow distribution in the nodes, see Section 2.3. When B is aperiodic, it can be calculated from the power method where the iteration $\left(\boldsymbol{\pi}^{t+1}\right)^T = \left(\boldsymbol{\pi}^t\right)^T B$ with some initial vector $\boldsymbol{\pi}^0$ tends to $\boldsymbol{\pi}$. The entry $\boldsymbol{\pi}_i^t$ represents then the proportion of flow in i at time t.

To define the TrustFlow vector, we will consider $\boldsymbol{x}(\mathcal{B})_i^t$ that represents the proportion of flow in i with blacklist $\mathcal{B}$ at time t. This is related to Eq. (6.4) but for a proportion of flow instead of a probability of presence. Then, using the flow matrix B, the flow with blacklist $\mathcal{B}$ coming in i by following a link is given by

$$\bar{\boldsymbol{x}}(\mathcal{B})_i^{t+1} = \sum_{j \to i : i \notin \mathcal{B}} \boldsymbol{x}(\mathcal{B})_j^t B_{ji},$$

for all $i \in \mathcal{N}$ and $\mathcal{B} \subset \mathcal{N}$. Let us remark that if $i \in \mathcal{B}$, the flow $\boldsymbol{x}(\mathcal{B})_j^t B_{ji}$ will not visit i, but he will rather jump according to $\boldsymbol{z}$ with a new blacklist. The total flow jumping at time $t+1$ is given by

$$\gamma^{t+1} = \sum_{j \to i : i \in \mathcal{B}} \boldsymbol{x}(\mathcal{B})_j^t B_{ji},$$

therefore, the proportion of flow reaching i after jumping is

$$\hat{\boldsymbol{x}}(\mathcal{B})_i^{t+1} = \gamma^{t+1} \boldsymbol{z}_i,$$

with $\mathcal{B} = \emptyset$ since the blacklist is emptied before jumping. The flow represented by the entries $\bar{\boldsymbol{x}}(\mathcal{B})_i^{t+1}$ and $\hat{\boldsymbol{x}}(\mathcal{B})_i^{t+1}$ have not yet taken into account the set $\mathcal{D}_i$ of negative links of i, therefore

$$\tilde{\boldsymbol{x}}(\mathcal{B})_i^{t+1} = \sum_{\bar{\mathcal{B}} : \mathcal{B} = \bar{\mathcal{B}} \cup \mathcal{D}_i} \bar{\boldsymbol{x}}(\bar{\mathcal{B}})_i^{t+1} + \hat{\boldsymbol{x}}(\bar{\mathcal{B}})_i^{t+1}$$

for all $i \in \mathcal{N}$ and $\mathcal{B} \subset \mathcal{N}$. We then obtain the proportions $\boldsymbol{x}(\mathcal{B})_i^{t+1}$ by normalizing every entry $\tilde{\boldsymbol{x}}(\mathcal{B})_i^{t+1}$ with $\mu_t = \sum_{i \in \mathcal{N}, \mathcal{B} \subset \mathcal{N}} \tilde{\boldsymbol{x}}(\mathcal{B})_i^{t+1}$. Hence, we can calculate the flow distribution in the nodes given by

$$\boldsymbol{x}^t = \sum_{\mathcal{B} \subset \mathcal{N}} \boldsymbol{x}(\mathcal{B})^t,$$

and the FlowTrust vector, with some zapping factor $1-c$, defined by

$$\boldsymbol{\pi} = (1-c)\boldsymbol{z} + (1-c) \sum_{t=1}^{\infty} c^t \boldsymbol{x}^t.$$

The simplified FlowTrust is based on the same assumption than for the s-PageTrust (Assumption **B**) and it also uses the vectors

$$\boldsymbol{x}(k)^t = \sum_{\mathcal{B} : k \in \mathcal{B}} \boldsymbol{x}(\mathcal{B})^t,$$

for all $k \in \mathcal{N}$. Then, following a similar development than in Section 6.3, we define the time-varying flow matrix

$$B_t = B \circ \left[\boldsymbol{\lambda}(1)^t \cdots \boldsymbol{\lambda}(n)^t \right],$$

where the vectors $\boldsymbol{\lambda}(k)^t$, for $k \in \mathcal{N}$, are defined as in Eq. (6.7). Hence, the sequences $(\boldsymbol{x}^t)$ and $(\boldsymbol{x}(k)^t)$ are given by

$$\begin{aligned} \left[\boldsymbol{x}^{t+1}\right]^T &= \frac{1}{\mu_t}\left[\boldsymbol{x}^t\right]^T B_t + \frac{\gamma^{t+1}}{\mu_t}\boldsymbol{z}^T, \\ \boldsymbol{x}(k)_i^{t+1} &= \begin{cases} \boldsymbol{x}_i^{t+1} & \text{if } (i,k) \in \mathcal{L}^-, \\ \frac{1}{\mu_t}\left(\left[\boldsymbol{x}^t\right]^T B_t\right)_i & \text{else,} \end{cases} \end{aligned}$$

with μ_t such that $\mathbf{1}^T \boldsymbol{x}^{t+1} = 1$, and

$$\gamma^{t+1} = \sum_{j \to k} \boldsymbol{x}(k)_j^t B_{jk}.$$

6.6 Conclusions

Results. In this chapter we described a natural extension of the random walks when a network contains negative links. That new walk, labeled the trust walk, needed to be simplified for large networks. This leads to the simplified trust walk for which the ranks are given by an iterative method that linearly converges with a rate depending on the choice of the zapping factor. We also introduce the degree of conviction β that allows us to change the effect of negative links on the ranking vectors. We then illustrate some interesting properties of the s-PageTrust, e.g., sensitivity and robustness, and the difference with the PageTrust on random networks. The results shows that the simplifications hardly modify the ranks given by the trust walk. We then use the large dataset of Epinions to emphasize the effect of negative links on the final ranking list when we compare the simplified trust walk with the PageRank. Finally, we discuss the other variants and extensions with the introduction of the FlowTrust and s-FlowTrust useful in the larger context of flow methods where the transition matrix S can be any nonnegative matrix B.

Future research. The analysis of the four algorithms that we introduced for other networks and with other nonnegative matrices can help to understand and validate these trust methods. Another point is to find alternative methods to rank the nodes of a network containing negative links. So far, there are few such methods as compared to ranking methods for networks with positive links. One of the main difficulties in these methods lies in their robustness when attacks are possible in the network.

Finally, in the variants when the blacklist is not emptied before jumping and zapping, the calculation of the associated ranking vectors and the issue of convergence still need some further analysis. It would be interesting to find a method usable for large networks, this maybe require some extra assumptions.

Chapter 7

Iterative Filtering in Voting Systems

In this chapter, a class of **voting systems** based on some iterative filtering is presented. These systems update the reputations of $n + p$ items, n objects and p raters, by applying some filter to the votes. Each rater evaluates a subset of objects leading to an $n \times p$ rating matrix with a given sparsity pattern. From this rating matrix a formula is defined for the reputation of raters and objects. Typically, we want to measure the credibility of the votes to then take them into account for the reputations of the objects.

We propose a natural and intuitive nonlinear formula, based on the variances of the votes, that provides an iterative algorithm labeled **quadratic Iterative Filtering**. That method linearly converges to the unique vector of reputations and this for any rating matrix. In contrast to classical outliers detection, no evaluation is discarded in this method but each one is taken into account with different weights for the reputations of the objects. The complexity of one iteration step is linear in the number of evaluations, making our algorithm efficient for large data set. Moreover, the method is suitable for dynamical votes and decentralized architecture.

Experiments show good robustness of the reputation of the objects against cheaters and spammers and good detection properties of cheaters and spammers.

7.1 Introduction

Motivation. The exponential growth of sites on the World Wide Web is accompanied by crucial needs of tools such as classification of documents, spam detection, traffic optimization, etc. In addition, we observe more and more use of interactive ratings from various raters. For example, evaluated items can be books on Amazon, movies on Movielens, videos on YouTube, computer scientists on Advogato, objects and raters on Epinions, or buyers and sellers on eBay. The list of such interactive sites is also constantly growing. In addition to these explicit evaluations, the simple fact to link to another web page is considered by most search engines – e.g., Google and Yahoo – as a positive evaluation. In these various forms of voting, all raters cannot be expected to be fully reliable or even honest. There is nothing to stop Movielens raters from giving random ratings to movies they have not even seen, or dishonest voters from giving biased opinions that favor their "friends".

From a commercial point of view, it is obvious that Web sites have a lot to gain from promoting confidence in such interactive rating systems. A famous example is given by Akerloff [1] in 1970, who pointed out the information asymmetry between the buyers and the sellers in the market for lemons. The buyers had more information than the sellers, making trading relationships less trustworthy. Transparency of the market could result from penalizing raters who give random or biased ratings. Another difficulty in a voting process concerns the biases due to influence between raters. For example in [26], the authors analyze the Eurovision contest and they claim that "the votes cast [...] are driven by linguistic and cultural proximities between singers and voting countries". Two questions ought to be addressed in this context:

1. How should the reputation of evaluated items be defined?

2. How can we measure the reliability of the raters?

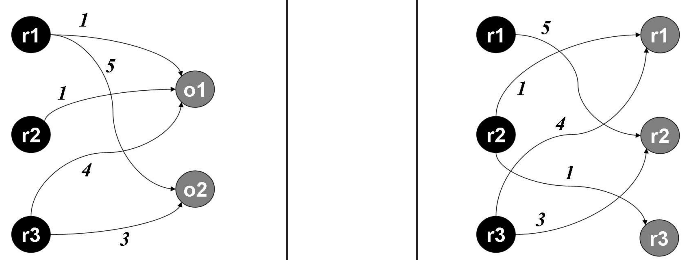
We distinguish here between the reputation of an item, i.e., what is generally said or believed about its character or standing, and the reliability of a rater, i.e., the probability that a rater will give a fair or relevant evaluation. We illustrate these definitions in the context of eBay (see the screen shot in Fig. 1.3 in Chapter 1): User Smith has a reputation that is simply equal to the percentage of positive votes that he receives.

That aggregated reputation, however, does not take into account the relevance of the votes given to Smith; the reliability of each rater needs to be taken into account.

Short review of voting systems. Many measures of reputation have been proposed these past years under the names of reputation, voting, ranking or trust systems and they deal with various contexts ranging from the classification of football teams to the reliability of each individual in peer to peer systems. Surprisingly enough, the most used method for reputation on the Web amounts simply to average the votes. In that case, the reputation is, for instance, the average of scores represented by five stars in YouTube, or the percentage of positive transactions in eBay. Therefore such a method trusts evenly each rater of the system. Besides this method, many other algorithms exploit the structure of networks generated by the votes: raters and evaluated items are nodes connected by votes as illustrated in Fig. 7.1. A great part of these methods use efficient eigenvector based techniques or trust propagation over the network to obtain the reputation of every node [29, 42, 76, 83, 85, 89, 99]. They can be interpreted as a distribution of some reputation flow over the network where reputations satisfy some transitivity: you have a high reputation if you have several incoming links coming from nodes with a high reputation. The averaging method, the eigenvector based techniques and trust propagation may suffer from noise in the data and bias from dishonest raters. For this reason, they are sometimes accompanied by statistical methods for spam detection [106, 52], like webpages trying to boost their PageRank scores by adding artificial incoming links [30, 7], or to measure the credibility of the raters by statistical models [16, 62, 87, 105]. Detected spam can then be simply removed from the data.

We describe the three main strategies for voting systems: simple methods averaging votes where raters are evenly trusted, eigenvector based techniques and trust propagation where reputations directly depend on reputations of the neighbors, and finally statistical measures to classify and possibly remove some of the items. The statistical method proposed by Laureti et al. in [62, 105] is an iterative filtering system closely related to the methods we present in the sequel.

Iterative Filtering systems. As explained in the previous paragraph, most methods do not compare the evaluations of the raters to



$$\begin{bmatrix} 1 & 1 & 4 \\ 5 & . & 3 \end{bmatrix} \qquad \begin{bmatrix} . & 1 & 4 \\ 5 & . & 1 \\ . & 1 & . \end{bmatrix}$$

Figure 7.1: Networks and matrices of votes from raters to items and between raters, in both cases it can be represented as a bipartite network.

deduce some weights of trust for these votes and then update the reputations of each item accordingly. Obviously the choice of a specific reputation system, with specific interpretations of the votes, depends on subjective properties that we just accept. For example, in the averaging method, we agree that every rater is taken into account in the same manner. In the PageRank algorithm, we accept that a random walk over the network is a good model of real navigation for a web surfer. In trust propagation over networks, we accept the transitivity of trust: if A trusts B and B trusts C, then A will trust C.

Concerning the IF systems, we will make the following assumption

Raters diverging often from other raters' opinion
are less taken into account.

We label this the *IF*-property and will formally define it later on. This property is at the heart of the filtering process and implies that all votes are taken into account, but with a continuous validation scale, in contrast with the direct deletion of outliers. Moreover, the weight of each rater depends on the distance between his votes and the reputation of the objects he evaluates: typically weights of random raters and outliers decrease during the iterative filtering. The main criticism one can have about the *IF*-property is that it discriminates “marginal” evaluators, i.e., raters who vote differently from the average opinion for many objects. However, IF systems may have different basins of attraction, each corresponding to a group of people with a coherent opinion.

Votes, raters and objects can appear, disappear or change making the system dynamical. This is for example the case when we consider a stream of news like in [19]: news sources and articles are ranked according to their publications over time. Nowadays, most sites driven by raters involve dynamical opinions. For instance, the blogs, the site Digg and the site Flickr are good places to exchange and discuss ideas, remarks and votes about various topics ranging from political election to photos and videos. We will see that IF systems allow to consider evolving voting matrices and then provide time varying reputations.

Structure. The first section introduces the definitions of IF systems and our method illustrated by a small example. Then the second section proves several convergence results for our method and some properties of the solution. The third section discusses the choice of the discriminant function that allows us to penalize differently the raters according to their votes, and then compares several IF systems including the method of Laureti et al. in [62, 105]. The fourth part extends some results to sparse voting matrices and dynamical data. The fifth part presents simulations on two real data sets. The last part concludes and gives some perspectives for further research.

7.2 Definitions and properties

For the sake of clarity, we first consider the case where the votes are fixed, i.e., the voting matrix does not change over time, and all objects are evaluated by all raters, i.e., the voting matrix is dense. The dynamical case and the sparsity pattern for the voting matrix will be analyzed in Section 7.5.

With these assumptions, we present the main properties of IF systems and then we restrict ourselves to the natural case of quadratic IF systems where the reputations are given by a linear combination of the votes and the weights of the raters are based on the Euclidean distance between the reputations and the votes.

General Notations. Let $X \in \mathbb{R}^{n\times p}$ be the voting matrix, $\boldsymbol{r} \in \mathbb{R}^n$ be the reputation vector of the objects and $\boldsymbol{w} \in \mathbb{R}^p_{\geq 0}$ be the weight vector of the raters. The entry X_{ij} is the vote to object i given by rater j and the vector $\boldsymbol{x}_j$, the j^{th} column of X, represents the votes of rater j:

$$X = [\boldsymbol{x}_1 \ \dots \ \boldsymbol{x}_p] \,.$$

We will assume that the votes belong to the interval $[a, b]$, i.e., $X_{ij} \in [a, b]$.

The bipartite network formed by the objects, the raters and their votes (see Fig. 7.1) is represented by the $n \times p$ adjacency matrix A, i.e., $A_{ij} = 1$ if object i is evaluated by rater j, and 0 otherwise. For the sake of simplicity, we assume in this section that every object has been evaluated by all raters

$$A_{ij} = 1 \quad \text{for all } i, j. \tag{7.1}$$

The general case where the bipartite network is not necessarily complete will be handled later.

The belief divergence $\boldsymbol{d}_j$ of rater j is the normalized euclidian distance between his votes and the reputation vector $\boldsymbol{r}$,

$$\boldsymbol{d} = \frac{1}{n} \begin{pmatrix} \|\boldsymbol{x}_1 - \boldsymbol{r}\|_2^2 \\ \vdots \\ \|\boldsymbol{x}_p - \boldsymbol{r}\|_2^2 \end{pmatrix}. \tag{7.2}$$

Therefore Eq.(7.2) are quadratic equations in $\boldsymbol{r}$ and amount to consider an estimate of the variances of the votes for every rater according to a given reputation vector $\boldsymbol{r}$. Let us already remark that when the bipartite network is not complete, i.e., Eq. (7.1) is not satisfied, then the number of votes varies from one rater to another. Therefore the normalization of the belief divergence $\boldsymbol{d}$ in Eq.(7.2) will change depending on this number.

The reputation and filtering functions. Before introducing *quadratic IF systems*, we define the two basic functions of these systems:

(1) the reputation function $\quad F : \mathbb{R}^p \to \mathbb{R}^n : F(\boldsymbol{w}) = \boldsymbol{r},$

that gives the reputation vector depending on the weights of the raters and implicitly on the voting matrix X;

(2) the filtering function $\quad G : \mathbb{R}^n \to \mathbb{R}^p_{\geq 0} : G(\boldsymbol{r}) = \boldsymbol{w},$

that gives the nonnegative weight vector for the raters depending on the reputation vector and implicitly on the voting matrix X.

We formalize the so-called *IF-property* described in the introduction that claims that raters diverging often from the opinion of other raters are less taken into account. We will make the reasonable assumption that raters with identical belief divergence receive equal weights. Hence, we can write

$$G(\boldsymbol{r}) = \begin{bmatrix} g(\boldsymbol{d}_1) \\ \vdots \\ g(\boldsymbol{d}_p) \end{bmatrix}. \tag{7.3}$$

We call the scalar function g the discriminant function associated with G. Eq. (7.3) indicates that every rater has the same discriminant function g, but we could also consider personalized functions g_j penalizing differently the raters.

A filtering function G satisfies the *IF*-property if its associated discriminant function $g : \mathbb{R} \to \mathbb{R}$ is nonnegative and monotonically decreasing. Therefore, the *IF*-property merely implies that a decrease in belief divergence $\boldsymbol{d}_j$ for any rater j corresponds to a larger (or equal) weight $\boldsymbol{w}_j$. Three choices of function g are shown to have interesting properties

$$\begin{aligned} g(d) &= d^{-k}, \\ g(d) &= e^{-k\,d}, \\ g(d) &= 1 - k\,d. \end{aligned}$$

All discriminant function g are nonnegative and monotonically decreasing for positive k and therefore satisfy the IF-property. However k must be small enough to keep g positive in the last function $g(d) = 1 - k\,d$ and hence to avoid negative weights. Our method is based on that function that will be compared to other discriminant functions in Section 7.4.

Our method. In the sequel, we focus on *quadratic IF systems* where we fix the reputation function F. That leads to the two ranking vectors $\boldsymbol{r}$ and $\boldsymbol{w}$ satisfying a system of quadratic equations in $\boldsymbol{r}$ and $\boldsymbol{w}$. Then our method is a quadratic IF system with the filtering function G chosen from the discriminant function $g(d) = 1 - k\,d$.

In *quadratic IF systems*, the reputation function $F(\boldsymbol{w})$ is naturally given by taking the weighted average of the votes

$$F(\boldsymbol{w}) = X\boldsymbol{w}/\|\boldsymbol{w}\|_1. \tag{7.4}$$

For any nonnegative vector $\boldsymbol{w}$. Since we assumed that the votes are in $[a, b]$, we then have that the reputation vector $\boldsymbol{r}$ belongs to the hypercube

$$\mathcal{H} := [a, b]^n, \tag{7.5}$$

and more precisely that it belongs to the convex hull $\mathcal{P} \subseteq \mathcal{H}$ for the set of points $\{\boldsymbol{x}_j : j = 1, \ldots, p\}$, that is,

$$\mathcal{P} = \{\boldsymbol{r} \in \mathbb{R}^n \,|\, \boldsymbol{r} = \sum_{j=1}^{p} \boldsymbol{w}_j \boldsymbol{x}_j \text{ with } \sum_{j=1}^{p} \boldsymbol{w}_j = 1 \text{ and } \boldsymbol{w}_j \geq 0\}. \tag{7.6}$$

Then the definition of *quadratic IF systems* follows.

Definition 5. Quadratic IF systems are systems of equations in the reputations $\boldsymbol{r}^t$ of the objects and the weights $\boldsymbol{w}^t$ of the raters that evolve over discrete time t according to the voting matrix X

$$\begin{aligned} \boldsymbol{r}^{t+1} &= F(\boldsymbol{w}^t) = X\boldsymbol{w}^t / \|\boldsymbol{w}^t\|_1, & (7.7)\\ \boldsymbol{w}^{t+1} &= G(\boldsymbol{r}^{t+1}), & (7.8) \end{aligned}$$

for some initial vector of weights $\boldsymbol{w}^0$.

Then, our method uses the affine function $g(d) = 1 - k\,d$ for the filtering function G. That leads to the following definition.

Definition 6. Our method is a quadratic IF systems with affine discriminant function $g(d) = 1 - k\ d$ for some positive k with

$$\begin{aligned} \boldsymbol{r}^{t+1} &= X\boldsymbol{w}^t / \|\boldsymbol{w}^t\|_1, & (7.9)\\ \boldsymbol{w}^{t+1} &= \mathbf{1} - k\,\frac{1}{n} \begin{pmatrix} \|\boldsymbol{x}_1 - \boldsymbol{r}^{t+1}\|_2^2 \\ \vdots \\ \|\boldsymbol{x}_p - \boldsymbol{r}^{t+1}\|_2^2 \end{pmatrix}. & (7.10) \end{aligned}$$

starting with equal weights $\boldsymbol{w}^0 = \mathbf{1}$.

We will show in the next section that quadratic IF systems corresponds to taking the steepest descent direction and to minimizing a particular energy function. Moreover, our method (with a simple condition on k) is guaranteed to converge to the unique minimum of the energy function.

Toy example. In order to illustrate our method, we show the sequences $(\boldsymbol{r}^t)$ and $(\boldsymbol{w}^t)$ for a given voting matrix X and their limit points $\boldsymbol{r}^*$ and $\boldsymbol{w}^*$. The context of ice skating for this example refers to the scandal occurring during the 2002 Olympic Winter Games. There, a French rater was criticized for having favored one of the skaters. We propose the following fictive voting matrix where every vote belongs to the interval $[0, 5]$:

$$X = \begin{bmatrix} 3.3 & 3.4 & 4.9 \\ 4.2 & 4.5 & 2.8 \end{bmatrix},$$

where the two rows correspond to the votes given to two ice skaters and the three columns represent the three raters. Obviously, the third rater tries to favor the first ice skater. If equal weights are given to the raters, we obtain the average votes: 3.87 for the first skater and 3.83 for the second one.

The iteration steps given by Eq. (7.9,7.10) with $k = 1/5$, that is $g(d) = 1 - d/5$, are

$$\begin{array}{ccccc} \boldsymbol{r}^1 & \boldsymbol{r}^2 & \boldsymbol{r}^3 & & \boldsymbol{r}^* \\ \begin{bmatrix} 3.87 \\ 3.83 \end{bmatrix} & \begin{bmatrix} 3.81 \\ 3.89 \end{bmatrix} & \begin{bmatrix} 3.79 \\ 3.91 \end{bmatrix} & \cdots & \begin{bmatrix} 3.79 \\ 3.91 \end{bmatrix} \end{array}$$

$$\begin{array}{ccccc} \boldsymbol{w}^1 & \boldsymbol{w}^2 & \boldsymbol{w}^3 & & \boldsymbol{w}^* \\ \begin{bmatrix} .95 \\ .93 \\ .79 \end{bmatrix} & \begin{bmatrix} .97 \\ .94 \\ .76 \end{bmatrix} & \begin{bmatrix} .97 \\ .95 \\ .75 \end{bmatrix} & \cdots & \begin{bmatrix} .97 \\ .95 \\ .75 \end{bmatrix} \end{array}$$

We see that the order is already reversed after one iteration and that the third weight is decreasing during the iterations. Let us remark that a larger k will penalize more severely the third rater; on the other hand a too large k could generate negative weights which is not allowed by the *IF*-property.

7.3 Convergence properties of our method

In this section, we mainly focus on several propositions related to the convergence properties of our method. The main results show that every iteration step $\boldsymbol{r}^t$ of our method in the system (7.9,7.10) corresponds to taking the steepest descent direction of a particular energy function

(Proposition 8) that has a unique stationary point in $\mathcal{H}$ (Proposition 9). The iteration steps converge to the unique minimum $\boldsymbol{r}^*$ under some condition on the parameter k (Theorem 8) with a q-linear rate of convergence (Proposition 12).

The energy function. The next proposition establishes the correspondence between the iteration steps of quadratic IF systems – and therefore of our method too – and steepest descent methods minimizing some energy function. The fixed points in Eq.(7.7,7.8) are then the stationary points of that energy function.

By using the trivial relation $\|\boldsymbol{w}\|_1 = \mathbf{1}^T\boldsymbol{w}$, we can reformulate one iteration step of $\boldsymbol{r}^t$ for a quadratic IF system as

$$\boldsymbol{r}^{t+1}(\mathbf{1}^T\boldsymbol{w}^t) = X\boldsymbol{w}^t, \tag{7.11}$$

hence a fixed point $\boldsymbol{r}^*$ is given by quadratic equations in $\boldsymbol{r}^*$ and $\boldsymbol{w}^*$,

$$\boldsymbol{r}^*(\mathbf{1}^T\boldsymbol{w}^*) = X\boldsymbol{w}^*, \tag{7.12}$$

where $\boldsymbol{w}^* = G(\boldsymbol{r}^*)$. Hence, the fixed points in Eq. (7.12) are the roots of the function

$$D(\boldsymbol{r}) = \frac{2}{n}(\boldsymbol{r}\mathbf{1}^T - X)\cdot G(\boldsymbol{r}), \tag{7.13}$$

that is the gradient of some scalar function introduced in the next proposition and labeled the energy function E.

Proposition 8. *The fixed points of quadratic IF systems with integrable discriminant function g, are the stationary points of the energy function*

$$E(\boldsymbol{r}) = \sum_{j=1}^{p}\int_0^{\boldsymbol{d}_j(\boldsymbol{r})} g(u)\,du + c, \tag{7.14}$$

where $\boldsymbol{d}_j$ is the belief divergence of rater j that depends on $\boldsymbol{r}$, and $c \in \mathbb{R}$ is a constant. Moreover one iteration step in quadratic IF systems corresponds to a steepest descent direction with a particular step size

$$\boldsymbol{r}^{t+1} = \boldsymbol{r}^t - \alpha^t\nabla_{\boldsymbol{r}}E(\boldsymbol{r}^t), \tag{7.15}$$

with $\alpha^t = \frac{n}{2\,\|\boldsymbol{w}^t\|_1}$.

Proof: We have $\nabla_{\boldsymbol{r}} E(\boldsymbol{r}) = \nabla_{\boldsymbol{r}} \boldsymbol{d}^T \cdot \nabla_{\boldsymbol{d}} E(\boldsymbol{r})$ with

$$\begin{aligned} \nabla_{\boldsymbol{r}} \boldsymbol{d}^T &= -\frac{2}{n}(X - \boldsymbol{r}\mathbf{1}^T), \\ \nabla_{\boldsymbol{d}} E(\boldsymbol{r}) &= G(\boldsymbol{r}). \end{aligned}$$

Therefore a stationary point $\boldsymbol{r}^*$ in E satisfies

$$\begin{aligned} -\frac{2}{n}(X - \boldsymbol{r}^*\mathbf{1}^T) \cdot G(\boldsymbol{r}^*) &= 0, \\ (X - \boldsymbol{r}^*\mathbf{1}^T)\boldsymbol{w}^* &= 0, \\ \boldsymbol{r}^*(\mathbf{1}^T \boldsymbol{w}^*) &= X\boldsymbol{w}^*, \end{aligned}$$

which corresponds to the fixed point equations given in Eq.(7.12).

We also have $\nabla_{\boldsymbol{r}} E(\boldsymbol{r}^t) = \nabla_{\boldsymbol{r}} (\boldsymbol{d}(\boldsymbol{r}^t))^T \cdot \nabla_{\boldsymbol{d}} E(\boldsymbol{r}^t)$ with

$$\begin{aligned} \nabla_{\boldsymbol{r}} (\boldsymbol{d}(\boldsymbol{r}^t))^T &= -\frac{2}{n}(X - \boldsymbol{r}^t\mathbf{1}^T), \\ \nabla_{\boldsymbol{d}} E(\boldsymbol{r}^t) &= G(\boldsymbol{r}^t) = \boldsymbol{w}^t. \end{aligned}$$

Therefore

$$\begin{aligned} \nabla_{\boldsymbol{r}} E(\boldsymbol{r}^t) &= -\frac{2}{n}(X - \boldsymbol{r}^t\mathbf{1}^T)\boldsymbol{w}^t \qquad (7.16) \\ &= -\frac{2}{n}(\mathbf{1}^T \boldsymbol{w}^t)(\boldsymbol{r}^{t+1} - \boldsymbol{r}^t) \\ &= -\frac{1}{\alpha^t}(\boldsymbol{r}^{t+1} - \boldsymbol{r}^t). \end{aligned}$$

∎

Remark. Proposition 8 can be easily extended to personalized discriminant functions g_j for $j = 1, \cdots, p$, by considering the following energy function E

$$E(\boldsymbol{r}) = \sum_{j=1}^{p} \int_0^{\boldsymbol{d}_j(\boldsymbol{r})} g_j(u)\, du + c, \qquad (7.17)$$

where c is any constant in $\mathbb{R}$

The stable fixed points of quadratic IF system minimize the sum of the integrals $\int_0^{\boldsymbol{d}_j} g(u)\, du$ on $j = 1, \ldots, p$, meaning that they minimize the sum of surfaces below g in the intervals $[0, \boldsymbol{d}_j]$ for $j = 1, \ldots, p$. For example, when g is constant, the weights are always equal and the unique fixed point is given by the average of the votes minimizing $\|\boldsymbol{d}\|_1$.

The energy function in Eq.(7.14) for the constant $c = n/2k$ associated with our method in Eq. (7.9,7.10) is then given by

$$E(\boldsymbol{r}) = -\frac{1}{2\,k}\,\boldsymbol{w}^T\boldsymbol{w}, \tag{7.18}$$

where $\boldsymbol{w}$ depends on $\boldsymbol{r}$ according the function $G(\boldsymbol{r})$, i.e.,

$$w = \mathbf{1} - k\,\frac{1}{n}\begin{pmatrix} \|\boldsymbol{x}_1 - \boldsymbol{r}\|_2^2 \\ \vdots \\ \|\boldsymbol{x}_p - \boldsymbol{r}\|_2^2 \end{pmatrix}.$$

Therefore, the energy function is a fourth-order polynomial in the variables $\boldsymbol{r}_i$, $i = 1, \cdots, n$,

$$E(\boldsymbol{r}) = -\frac{1}{2\,k}\sum_{j=1}^{p}\left(1 - k\,\frac{1}{n}\|\boldsymbol{x}_j - \boldsymbol{r}\|_2^2\right)^2. \tag{7.19}$$

We will see later on that this energy function decreases with the iteration steps, i.e., the sequence $(E(\boldsymbol{r}^t))$ is monotonically decreasing, and under some assumption on k, it converges to the unique minimum.

Uniqueness. The following proposition proves that the stable point of our method is unique, under some condition on parameter k. We will consider the set of admissible k to guarantee that the weights $\boldsymbol{w} = G(\boldsymbol{r})$ remain positive for every possible reputation vector $\boldsymbol{r}$ in the hypercube $\mathcal{H}$, that is given by

$$\mathcal{K} = \{k \in \mathbb{R}_{\geq 0} : \mathbf{1} - k\,\frac{1}{n}\begin{pmatrix} \|\boldsymbol{x}_1 - \boldsymbol{r}\|_2^2 \\ \vdots \\ \|\boldsymbol{x}_p - \boldsymbol{r}\|_2^2 \end{pmatrix} > 0 \text{ for all } \boldsymbol{r} \in \mathcal{H}\}. \tag{7.20}$$

Moreover the result in the next proposition follows directly from the nature of the energy function E that gives several conditions on the existence of stationary points. These conditions are exposed in the following lemma.

Lemma 7. *Let the function $E : \mathbb{R}^n \to \mathbb{R} : E(\boldsymbol{r}) = z$ be a fourth-order polynomial and let $\mathcal{H}$ be some hypercube in $\mathbb{R}^n$. If*

$$\lim_{\|\boldsymbol{r}\|\to\infty} E(\boldsymbol{r}) = -\infty$$

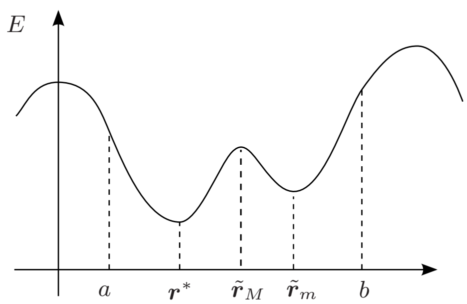


Figure 7.2: (a) if $\boldsymbol{r}^*$ is a strict minimum, then another minimum $\tilde{\boldsymbol{r}}_m$ leads to a contradiction; (b) if $\tilde{\boldsymbol{r}}_M$ is a strict maximum, then there would be two other maxima (before a and after b) leading to a contradiction.

and the steepest descent direction on the boundary of $\mathcal{H}$ points strictly inside $\mathcal{H}$, then E has a unique stationary point in $\mathcal{H}$ which is a strict minimum.

Proof: Since the steepest descent on the boundary of $\mathcal{H}$ points strictly inside $\mathcal{H}$, there is no stationary point on the boundary and there is at least one strict minimum in $\text{int}(\mathcal{H})$ that we label $\boldsymbol{r}^*$. Let prove that the existence of another stationary point leads to a contradiction with the hypothesis.

Let us first assume that $\tilde{\boldsymbol{r}}_m$ is another minimum of E (strict or not). The line passing by the two points $\boldsymbol{r}^*$ and $\tilde{\boldsymbol{r}}_m$ is given by $\ell(y) = \boldsymbol{r}^* + y\,(\tilde{\boldsymbol{r}}_m - \boldsymbol{r}^*)$ and the restriction

$$e(y) := E \circ \ell(y) \tag{7.21}$$

is a polynomial of degree 4 with two minima in $y = 0$ and $y = 1$. But this is not possible with the hypothesis $\lim_{\|\boldsymbol{r}\|\to\infty} E(\boldsymbol{r}) = -\infty$, see Fig. 7.2(a).

Let us now assume that $\tilde{\boldsymbol{r}}_M$ is a maximum of E (strict or not). The line passing by $\tilde{\boldsymbol{r}}_M$ with the direction $\boldsymbol{e}_1$ is given by $\ell'(y) = \tilde{\boldsymbol{r}}_M + \boldsymbol{e}_1(y - \tilde{\boldsymbol{r}}_1)$. The restriction

$$e'(y) := E \circ \ell'(y) \tag{7.22}$$

would be a polynomial of degree 4 with three maxima: one in $]-\infty, 0[$, one in $y = (\tilde{\boldsymbol{r}}_M)_1$ and one in $]1, \infty[$, see Fig. 7.2(b).

Let us finally assume that $\tilde{\boldsymbol{r}}$ is a saddle point in $\mathcal{H}$. This implies that there is an increasing trajectory starting in $\tilde{\boldsymbol{r}}$ and following the steepest ascent directions. By the condition on the boundary, such a trajectory cannot escape from $\mathcal{H}$. Therefore, it should reach a maximum $\tilde{\boldsymbol{r}}_M$ in $\mathcal{H}$ which is impossible. ∎

Proposition 9. *If $k \in \mathcal{K}$, the system in Eq. (7.9,7.10) has a unique fixed point $\boldsymbol{r}^*$.*

Proof: First of all, let us remark that the reputation of an object that receives the same vote from every rater, i.e., the corresponding row of X has equal entries, will remain the same during the iteration steps. Therefore, we restrict ourselves to the reputation of the objects that receive at least two different votes, say that there are $\tilde{n}$ such objects. Let $\tilde{\boldsymbol{r}} \in \mathbb{R}^{\tilde{n}}$ be their reputation vector and $\tilde{X} \in \mathbb{R}^{\tilde{n}\times p}$ be their associated voting matrix. The iteration steps are then given by

$$\begin{aligned}
\tilde{\boldsymbol{r}}^{t+1} &= \tilde{X}\boldsymbol{w}^t/\|\boldsymbol{w}^t\|_1,\\
\tilde{\boldsymbol{w}}^{t+1} &= \mathbf{1} - k\,\frac{1}{n}\begin{pmatrix} \|\tilde{\boldsymbol{x}}_1 - \tilde{\boldsymbol{r}}^{t+1}\|_2^2 \\ \vdots \\ \|\tilde{\boldsymbol{x}}_p - \tilde{\boldsymbol{r}}^{t+1}\|_2^2 \end{pmatrix},
\end{aligned}$$

and since $k \in \mathcal{K}$ and $\tilde{\boldsymbol{w}}^0 = \mathbf{1}$, the vector $\tilde{\boldsymbol{r}}^0$ is in the hypercube $\tilde{\mathcal{H}} := [a,b]^{\tilde{n}}$ and the sequence $(\tilde{\boldsymbol{r}}^t)$ remains also in the hypercube $\tilde{\mathcal{H}}$.

Since now every object has at least two different votes in $[a,b]$ given by two raters with positive weights ($k \in \mathcal{K}$), we have that the sequence $(\tilde{\boldsymbol{r}}^t)$ remains in $\mathrm{int}(\tilde{\mathcal{H}})$. Then by Proposition 8, it is sufficient to show that the energy function $\tilde{E}(\tilde{\boldsymbol{r}}) = -\frac{1}{2\,k}\sum_{j=1}^{p}\left(1 - k\,\frac{1}{n}\|\tilde{\boldsymbol{x}}_j - \tilde{\boldsymbol{r}}\|_2^2\right)^2$. has a unique stationary point in $\mathrm{int}(\tilde{\mathcal{H}})$.

The steepest descent direction at any point $\tilde{\boldsymbol{r}}^t \in \tilde{\mathcal{H}}$ is given by Eq.(7.15),

$$-\nabla_{\tilde{\boldsymbol{r}}}\tilde{E}(\tilde{\boldsymbol{r}}^t) = \frac{1}{\alpha^t}(\tilde{\boldsymbol{r}}^{t+1} - \tilde{\boldsymbol{r}}^t),$$

and since for any point $\tilde{\boldsymbol{r}}^t$ on the boundary of $\tilde{\mathcal{H}}$, the next point $\tilde{\boldsymbol{r}}^{t+1}$ belongs to $\mathrm{int}(\tilde{\mathcal{H}})$ (all weights are strictly positive), the steepest descent direction of $\tilde{E}$ on the boundary of $\tilde{\mathcal{H}}$ points strictly inside $\tilde{\mathcal{H}}$. Therefore using Lemma 7 for $\tilde{E}$, there is a unique stationary point in $\tilde{\mathcal{H}}$ which is a minimum, and by Proposition 8, it is the unique fixed point of the system in Eq. (7.9,7.10). ∎

Fig. 7.3 illustrates the stationary points of E when k is taken larger. First, if $k = \sup\mathcal{K}$, then the weights $\boldsymbol{w}$ are nonnegative (rather than positive), therefore maxima and saddle points can appear on the boundary of $\mathcal{H}$ (Fig. 7.3(b) is close to this case). Therefore iterations have to

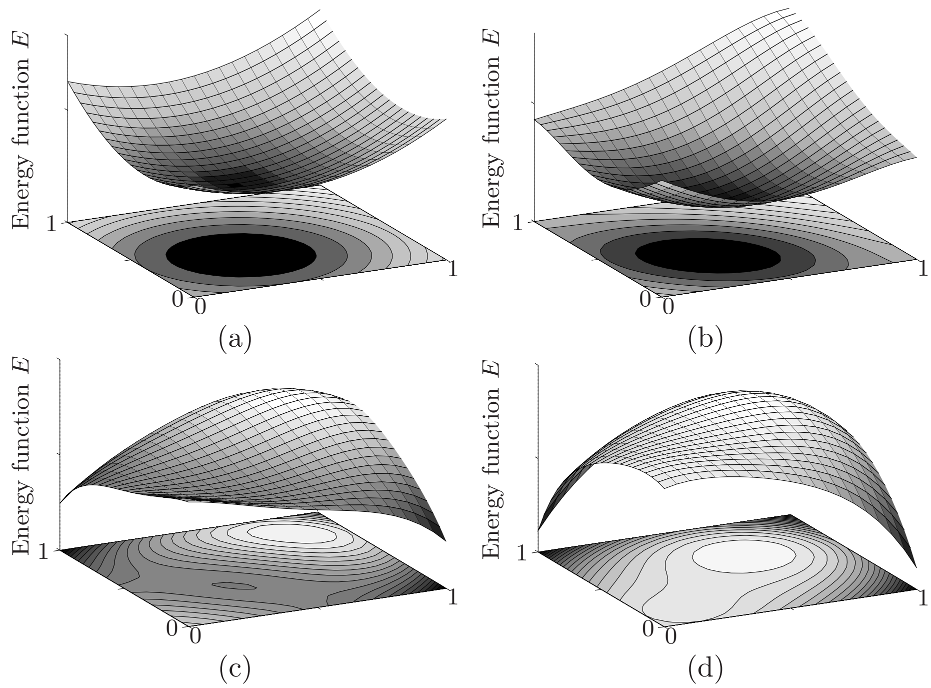


Figure 7.3: Four energy functions with two objects and increasing values of k. We have in the unit square: (a) a unique minimum; (b) a unique minimum but other stationary points are close to the boundary; (c) a unique minimum and other stationary points; (d) a unique maximum.

avoid these unstable points. Second, if k is strictly larger than $\sup \mathcal{K}$, then maxima can appear inside $\mathcal{H}$, see Fig. 7.3(c). Moreover, the existence of a minimum is not guaranteed anymore. However, if it exists, it remains unique and it can be show that its basin of attraction contain an open neighborhood of that point. This choice of larger k is discussed at the end of the section.

Convergence to $\boldsymbol{r}^*$. In this subsection, we prove the convergence of our method that reaches the minimum of the energy function E in $\mathcal{H}$ by taking the steepest descent direction at every iteration step. We also show convergence of an alternative method, where one updates one by one the entries of the reputation vector, which amounts to take a coordinate descent at every iteration step.

We can already remark that one iteration step of our method can be written as a particular minimization step on the function E.

Proposition 10. *The system (7.9,7.10) satisfies for all $t \geq 0$,*

$$\boldsymbol{r}^{t+1} = \arg\min_{r} \left[-\frac{1}{2\,k}\, G(\boldsymbol{r})^T G(\boldsymbol{r}^t) \right]. \tag{7.23}$$

Proof: It is sufficient to look at $\nabla_{\boldsymbol{r}} \left[-\frac{1}{2\,k}\, G(\boldsymbol{r})^T G(\boldsymbol{r}^t) \right]$ that is given by

$$\begin{aligned} -\frac{1}{2\,k}\, \nabla_{\boldsymbol{r}} G(\boldsymbol{r})^T \boldsymbol{w}^t &= -\frac{1}{2\,k}\, \nabla_{\boldsymbol{r}} \left(-k\, \boldsymbol{d}\right)^T \boldsymbol{w}^t \\ &= -\frac{1}{n}(X - \boldsymbol{r}\mathbf{1}^T)\boldsymbol{w}^t, \end{aligned}$$

which is zero only for $\boldsymbol{r} = \boldsymbol{r}^{t+1}$, and at the Hessian which is given by $\frac{\mathbf{1}^T \boldsymbol{w}^t}{n} I$ and is positive definite. Therefore $\boldsymbol{r}^{t+1}$ is the unique minimum. ∎

Therefore, we have for all t that $(\boldsymbol{w}^{t+1})^T(\boldsymbol{w}^t) \geq (\boldsymbol{w}^t)^T(\boldsymbol{w}^t)$. This is not sufficient to claim that the energy function decreases after every iteration step. We rather need to show that

$$(\boldsymbol{w}^{t+1})^T(\boldsymbol{w}^{t+1}) \geq (\boldsymbol{w}^t)^T(\boldsymbol{w}^t).$$

Before introducing the theorem that proves that the energy function decreases after every iteration step, we present the two following lemmas that are useful for that theorem. The first lemma will also be used further when we will consider a sparsity pattern in the voting matrix.

Lemma 8. *Given two matrices M and A such that $M \circ A = M$ we have*

$$[M^T - A^T \circ \mathbf{1}\boldsymbol{c}^T]^{\circ 2}\mathbf{1} = [M^T]^{\circ 2}\mathbf{1} - 2M^T\boldsymbol{c} + A^T\boldsymbol{c}^{\circ 2}.$$

Proof:

$$\begin{aligned} [M^T - A^T \circ \mathbf{1}\boldsymbol{c}^T]^{\circ 2}\mathbf{1} &= \left(A^T \circ [M^T - \mathbf{1}\boldsymbol{c}^T]^{\circ 2}\right)\mathbf{1} \\ &= \left(A^T \circ [(M^T)^{\circ 2} - 2M^T \circ \mathbf{1}\boldsymbol{c}^T + (\mathbf{1}\boldsymbol{c}^T)^{\circ 2}]\right)\mathbf{1} \\ &= \left((M^T)^{\circ 2} - 2M^T \circ \mathbf{1}\boldsymbol{c}^T + A^T \circ (\mathbf{1}\boldsymbol{c}^T)^{\circ 2}\right)\mathbf{1} \\ &= [M^T]^{\circ 2}\mathbf{1} - 2M^T\boldsymbol{c} + A^T\boldsymbol{c}^{\circ 2}. \end{aligned}$$

∎

Lemma 9. *If $k \in \mathcal{K}$, the sequence of weights $(\boldsymbol{w}^t)$ in the system (7.9,7.10) satisfies*

$$\mathbf{1}^T \boldsymbol{w}^t > 1$$

for all $t \geq 0$.

Proof: We have

$$\begin{aligned}
\min_{t\geq 0} \mathbf{1}^T \boldsymbol{w}^t &= \min_{t\geq 0} \mathbf{1}^T G(\boldsymbol{r}^t) \\
&\geq \min_{\boldsymbol{r}\in\mathcal{P}} \mathbf{1}^T G(\boldsymbol{r}) \\
&\geq \min_{\boldsymbol{r}\in\mathcal{P}} \mathbf{1}^T \mathbf{1} - \frac{k}{n} \mathbf{1}^T \begin{pmatrix} \|\boldsymbol{x}_1 - \boldsymbol{r}\|_2^2 \\ \vdots \\ \|\boldsymbol{x}_p - \boldsymbol{r}\|_2^2 \end{pmatrix} \\
&\geq p - \frac{k}{n} \max_{\boldsymbol{r}\in\mathcal{P}} \sum_{j=1}^{p} \|\boldsymbol{x}_j - \boldsymbol{r}\|_2^2.
\end{aligned}$$

Since the function $C(\boldsymbol{r}) := \sum_{j=1}^{p} \|\boldsymbol{x}_j - \boldsymbol{r}\|_2^2$ is convex, it has a unique maximum $\boldsymbol{r}^*$ at some vertex of the convex hull $\mathcal{P}$, i.e., $\boldsymbol{r}^* = \boldsymbol{x}_k$ for some $k \in \{1, \ldots, p\}$. Hence,

$$\begin{aligned}
\min_{t\geq 0} \mathbf{1}^T \boldsymbol{w}^t &\geq p - \frac{k}{n} \max_{\boldsymbol{r}\in\mathcal{P}} \sum_{j=1, j\neq k}^{p} \|\boldsymbol{x}_j - \boldsymbol{x}_k\|_2^2 \\
&= 1 + \sum_{j=1, j\neq k}^{p} 1 - \frac{k}{n} \|\boldsymbol{x}_j - \boldsymbol{x}_k\|_2^2 \\
&= 1 + \sum_{j=1, j\neq k}^{p} G(\boldsymbol{x}_k)_j \\
&> 1 \qquad \text{(the weights } \boldsymbol{w} = G(\boldsymbol{x}_k) \text{ are positive).}
\end{aligned}$$

■

Theorem 8. *If $k \in \mathcal{K}$, the system (7.9,7.10) converges to the unique fixed point $\boldsymbol{r}^* \in \mathcal{H}$.*

Proof: First, we show that the energy function E decreases between any two iterations, i.e., $E(\boldsymbol{r}^{t+1}) \leq E(\boldsymbol{r}^t)$ for all $t \geq 0$. This is equivalent to prove that $(\boldsymbol{w}^{t+1})^T(\boldsymbol{w}^{t+1}) \geq (\boldsymbol{w}^t)^T(\boldsymbol{w}^t)$. Let us express $\boldsymbol{w}^{t+1}$ in terms

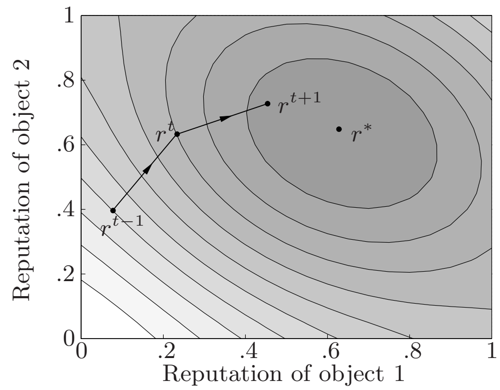


Figure 7.4: Two iteration steps by our method. Each one decreases the energy function E. They take the steepest descent direction and converge to the minimum $\boldsymbol{r}^*$.

of $\boldsymbol{w}^t$.

$$\begin{aligned}
\boldsymbol{w}^{t+1} &= \mathbf{1} - \frac{k}{n}[X^T - \mathbf{1}(\boldsymbol{r}^{t+1})^T]^{\circ 2}\mathbf{1} \\
&= \mathbf{1} - \frac{k}{n}[X^T - \mathbf{1}(\boldsymbol{r}^t)^T - \mathbf{1}(\boldsymbol{r}^{t+1} - \boldsymbol{r}^t)^T]^{\circ 2}\mathbf{1} \\
&\quad \text{(by Lemma 8 with } A = \mathbf{1}\mathbf{1}^T,\ M^T = X^T - \mathbf{1}(\boldsymbol{r}^t)^T \text{ and } \boldsymbol{c} = \boldsymbol{r}^{t+1} - \boldsymbol{r}^t) \\
&= \boldsymbol{w}^t + \frac{k}{n}\left(2\,(X^T - \mathbf{1}(\boldsymbol{r}^t)^T)(\boldsymbol{r}^{t+1} - \boldsymbol{r}^t) - (\boldsymbol{r}^{t+1} - \boldsymbol{r}^t)^T(\boldsymbol{r}^{t+1} - \boldsymbol{r}^t)\mathbf{1}\right) \\
&= \boldsymbol{w}^t + \frac{k}{n}\,\boldsymbol{q},
\end{aligned}$$

with $\boldsymbol{q} := \left(2\,(X^T - \mathbf{1}(\boldsymbol{r}^t)^T)(\boldsymbol{r}^{t+1} - \boldsymbol{r}^t) - (\boldsymbol{r}^{t+1} - \boldsymbol{r}^t)^T(\boldsymbol{r}^{t+1} - \boldsymbol{r}^t)\mathbf{1}\right)$. Hence,

$$\begin{aligned}
(\boldsymbol{w}^{t+1})^T(\boldsymbol{w}^{t+1}) &= (\boldsymbol{w}^t + \frac{k}{n}\,\boldsymbol{q})^T(\boldsymbol{w}^t + \frac{k}{n}\,\boldsymbol{q}) \\
&= (\boldsymbol{w}^t)^T(\boldsymbol{w}^t) + \frac{k^2}{n^2}\,\boldsymbol{q}^T\boldsymbol{q} + 2\,\frac{k}{n}\boldsymbol{q}^T\boldsymbol{w}^t.
\end{aligned}$$

Therefore, it is sufficient to show that $\boldsymbol{q}^T\boldsymbol{w}^t \geq 0$. This follows from

$$\begin{aligned}
\boldsymbol{q}^T\boldsymbol{w}^t &= 2\,(\boldsymbol{r}^{t+1} - \boldsymbol{r}^t)^T(X - \boldsymbol{r}^t\mathbf{1}^T)\boldsymbol{w}^t - (\boldsymbol{r}^{t+1} - \boldsymbol{r}^t)^T(\boldsymbol{r}^{t+1} - \boldsymbol{r}^t)\mathbf{1}^T\boldsymbol{w}^t \\
&= 2\,(\boldsymbol{r}^{t+1} - \boldsymbol{r}^t)^T(\boldsymbol{r}^{t+1} - \boldsymbol{r}^t)\mathbf{1}^T\boldsymbol{w}^t - (\boldsymbol{r}^{t+1} - \boldsymbol{r}^t)^T(\boldsymbol{r}^{t+1} - \boldsymbol{r}^t)\mathbf{1}^T\boldsymbol{w}^t \\
&= \|\boldsymbol{r}^{t+1} - \boldsymbol{r}^t\|_2^2\,\mathbf{1}^T\boldsymbol{w}^t, \qquad (7.24)
\end{aligned}$$

which is greater than 0 since $\boldsymbol{w}^t$ is a positive vector because of the condition on k. This shows that the energy is strictly decreasing when

$\boldsymbol{r}^{t+1} \neq \boldsymbol{r}^t$ with

$$E(\boldsymbol{r}^{t+1}) - E(\boldsymbol{r}^t) \leq -\frac{1}{n} \|\boldsymbol{r}^{t+1} - \boldsymbol{r}^t\|_2^2, \tag{7.25}$$

where we use $\|\boldsymbol{w}^t\|_1 > 1$ by Lemma 9. Since E is lower bounded in $\mathcal{H}$, the sequence $(\boldsymbol{r}^t)$ converges to a single limit point $\boldsymbol{r}^* \in \mathcal{H}$. Then it follows from Eq.(7.15) that

$$\|\nabla_{\boldsymbol{r}} E(\boldsymbol{r}^t)\|_2 = \frac{2\,\|\boldsymbol{w}^t\|_1}{n} \|\boldsymbol{r}^{t+1} - \boldsymbol{r}^t\|_2 \leq \frac{2p}{n} \|\boldsymbol{r}^{t+1} - \boldsymbol{r}^t\|_2,$$

where we used $\|\boldsymbol{w}^t\|_1 = \mathbf{1}^T\mathbf{1} - \mathbf{1}^T\boldsymbol{d}(\boldsymbol{r}^t) \leq p$. Therefore at $\boldsymbol{r}^*$, the gradient is zero and by Proposition 9, the point $\boldsymbol{r}^*$ is the unique fixed point. ■

The system (7.9,7.10) can be modified to update a single reputation $\boldsymbol{r}_i$ at a time. Using the function F and G of our method, we have the following updates

$$\boldsymbol{r}_i^{t+1} = F_i(\boldsymbol{w}) \quad \text{and} \quad r_q^{t+1} = r_q^t \qquad \text{for } q \neq i, \tag{7.26}$$

$$\boldsymbol{w}^{t+1} = G(\boldsymbol{r}^{t+1}), \tag{7.27}$$

and the coordinate i is incremented with t such that $i = 1 + t\text{modulo}n$ and we need n iteration steps to update all the entries of $\boldsymbol{r}^t$ and to obtain $\boldsymbol{r}^{t+n}$. This method can advantageously be applied to distributed voting systems where the votes are not centralized. Moreover if the voting matrix is sparse, then updates in the system (7.26,7.27) become cheaper.

These equations provide a coordinate descent method on the energy function E with a particular step size. Indeed, it can be shown that one iteration step is given by

$$\boldsymbol{r}_i^{t+1} = \boldsymbol{r}_i^t - \alpha^t \frac{\partial E(\boldsymbol{r}^t)}{\partial \boldsymbol{r}_i} \boldsymbol{e}_i,$$

and similarly to the previous method, the convergence to the minimum of E is guaranteed.

Proposition 11. *If $k \in \mathcal{K}$, the system (7.26,7.27) converges to the unique fixed point in $\boldsymbol{r}^* \in \mathcal{H}$.*

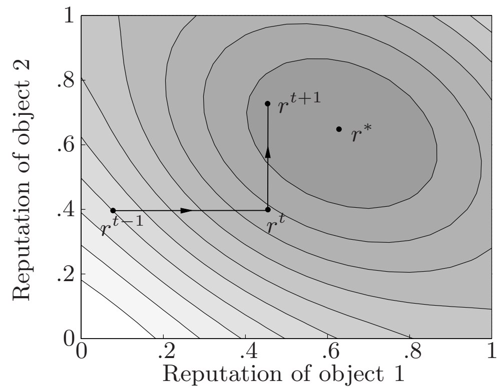


Figure 7.5: Two iteration steps of the system (7.26,7.27). Each one decreases the energy function E, they take some coordinate descent direction and converges to the minimum $\boldsymbol{r}^*$.

Proof: We still have that the energy function E decreases with the iterations, and this is proved by considering the same development as in the proof of Theorem 8 until Eq. (7.24) for $\boldsymbol{q}^T\boldsymbol{w}^t$. Without loss of generality, we can assume $\boldsymbol{r}^{t+1} - \boldsymbol{r}^t = (r_1^{t+1} - r_1^t)\boldsymbol{e}_1$ meaning that at time t we begin a new series of updates on the entries of the vector $\boldsymbol{r}$. Hence,

$$\boldsymbol{q}^T\boldsymbol{w}^t = (r_1^{t+1} - r_1^t)^2\mathbf{1}^T\boldsymbol{w}^t,$$

which is greater than 0 since $\boldsymbol{w}^t$ is a positive vector because of the condition on k. This shows that the energy is strictly decreasing when $r_1^{t+1} \neq r_1^t$ with, similarly to Eq.(7.25),

$$E(\boldsymbol{r}^{t+1}) - E(\boldsymbol{r}^t) \leq -\frac{1}{n}\,(r_1^{t+1} - r_1^t)^2,$$

and by summing the terms, we obtain

$$E(\boldsymbol{r}^{t+n}) - E(\boldsymbol{r}^t) \leq -\frac{1}{n}\,\|\boldsymbol{r}^{t+n} - \boldsymbol{r}^t\|_2^2.$$

Since E is lower bounded in $\mathcal{H}$, this implies that the sequence $(\boldsymbol{r}^t)$ converges to a single limit point $\boldsymbol{r}^* \in \mathcal{H}$. Finally, it follows from Eq.(7.15) that

$$\begin{aligned}\left|\frac{\partial E(\boldsymbol{r}^t)}{\partial \boldsymbol{r}_1}\right| &= \frac{2\,\|\boldsymbol{w}^t\|_1}{n}\,|r_1^{t+1} - r_1^t| \\ &\leq \frac{2p}{n}|r_1^{t+1} - r_1^t|,\end{aligned}$$

where we used that $\|\boldsymbol{w}^t\|_1 = \mathbf{1}^T\mathbf{1} - \mathbf{1}^T\boldsymbol{d}(\boldsymbol{r}^t) \leq p$. Then by summing the terms, we obtain

$$\sqrt{\sum_{i=1}^{n} \left| \frac{\partial E(\boldsymbol{r}^{t+i-1})}{\partial \boldsymbol{r}_i} \right|^2} \leq \frac{2p}{n} \|\boldsymbol{r}^{t+n} - \boldsymbol{r}^t\|_2.$$

Therefore the gradient is zero at $\boldsymbol{r}^*$ and by Proposition 9, the point $\boldsymbol{r}^*$ is the unique fixed point. ■

Let us remark that there exist other methods to calculate the roots of a polynomial of degree 3. The two methods we just described have a reasonable rate of convergence – that will be shown q-linear in the next subsection –, and have the advantage to be directly usable for dynamical votes – as we will see in Section 7.5. In the context of dynamical votes, converging too fast in one iteration step would loose the dynamic aspect of the votes where the new opinion should not be too far away from the previous one. For that purpose, it can be relevant to define a parameter that takes into account a memory effect on the votes. Let μ be this parameter in $[0, 1]$, then the iterations become

$$\boldsymbol{r}^{t+1} = \mu \boldsymbol{r}^t + (1-\mu) X \frac{\boldsymbol{w}^t}{\mathbf{1}^T \boldsymbol{w}^t},$$

and one updates $\boldsymbol{w}$ as before. The proofs of convergence of Theorem 8 and Proposition 11 can be easily extended to this case. We will see in the next subsection how increasing values of μ will decrease the rate of convergence of the method.

Rate of convergence. The following proposition claims that our method is locally convergent with a q-linear rate of convergence if there exists a minimum $\boldsymbol{r}^* \in \mathcal{H}$. Let us remark that this condition is weaker than the condition on k used in the previous propositions, i.e., $k \in \mathcal{K}$. Therefore, it is possible that our method converges for $k \notin \mathcal{K}$ depending on the initial vector $\boldsymbol{w}^0$. That will be discussed in the next subsection. After the proposition, we also give the impact of the parameter μ on the rate of convergence.

Proposition 12. *If the energy function E in Eq.(7.18) has a minimum, then the system (7.9,7.10) is locally convergent and its rate of convergence is q-linear.*

Proof: We will prove that the function of iteration $F \circ G(\boldsymbol{r})$ is a contraction mapping in some neighborhood $\mathcal{N}$ of the minimum $\boldsymbol{r}^*$ of E. For that purpose, it is sufficient to show that $\|\nabla_{\boldsymbol{r}} F \circ G(\boldsymbol{r}^*)\|_2 < 1$, see Section 2.5. This gradient is given by

$$\nabla_{\boldsymbol{r}} \left(F \circ G(\boldsymbol{r}^*)\right) = \frac{2k}{n\, \mathbf{1}^T \boldsymbol{w}^*}(X - \boldsymbol{r}^* \mathbf{1}^T)(X - \boldsymbol{r}^* \mathbf{1}^T)^T, \qquad (7.28)$$

where $\boldsymbol{w}^* = G(\boldsymbol{r}^*)$. Therefore, this gradient is positive semi definite when $\mathbf{1}^T \boldsymbol{w}^* > 0$ (not guaranteed anymore since the weights can be negative). On the other hand, the Hessian of E at $\boldsymbol{r}^*$ must be positive definite (otherwise it contradicts the hypothesis on $\boldsymbol{r}^*$ as a minimum). The Hessian is given by

$$\begin{aligned}
\nabla^2_{\boldsymbol{r}} E(\boldsymbol{r}^*) &= \frac{2\, \mathbf{1}^T \boldsymbol{w}^*}{n} \boldsymbol{I} - \frac{4k}{n^2}(X - \boldsymbol{r}^* \mathbf{1}^T)(X - \boldsymbol{r}^* \mathbf{1}^T)^T \\
&= \frac{2\, \mathbf{1}^T \boldsymbol{w}^*}{n} \left[\boldsymbol{I} - \nabla_{\boldsymbol{r}} \left(F \circ G(\boldsymbol{r}^*)\right)\right].
\end{aligned}$$

This implies that $\mathbf{1}^T \boldsymbol{w}^* > 0$ and the eigenvalues of $\frac{4k}{n^2}(X - \boldsymbol{r}^* \mathbf{1}^T)(X - \boldsymbol{r}^* \mathbf{1}^T)^T$ have to be strictly less than $\frac{2\, \mathbf{1}^T \boldsymbol{w}^*}{n}$. Therefore the spectrum of $\nabla_{\boldsymbol{r}} \left(F \circ G(\boldsymbol{r}^*)\right)$ belongs to $[0, 1[$ and hence its $2-$norm is strictly less than 1.

In order to find the asymptotic rate of convergence, we consider the error at time k given by $\boldsymbol{\epsilon}^k := \boldsymbol{r}^k - \boldsymbol{r}^*$. This error is given using the first order Taylor expansion at $\boldsymbol{r}^*$ by

$$\boldsymbol{\epsilon}^k = \nabla_{\boldsymbol{r}} \left(F \circ G(\boldsymbol{r}^*)\right) \boldsymbol{\epsilon}^{k-1} + O(\|\boldsymbol{\epsilon}^{k-1}\|^2).$$

Since the gradient in Eq.(7.28) is a symmetric matrix with all eigenvalues in $[0, 1[$, the asymptotic convergence is $q-$linear with a rate of convergence given by $\rho\left(\nabla_{\boldsymbol{r}} \left(F \circ G(\boldsymbol{r}^*)\right)\right)$ for the Euclidean norm, see Fig. 7.6. ∎

The same proposition can be studied for the system (7.26,7.27) with only minor changes in the proof. Moreover, if we consider the parameter μ, Eq. (7.28) becomes

$$\nabla_{\boldsymbol{r}} \left(F \circ G(\boldsymbol{r}^*)\right) = \mu I + (1 - \mu)\frac{2k}{n\, \mathbf{1}^T \boldsymbol{w}^*}(X - \boldsymbol{r}^* \mathbf{1}^T)(X - \boldsymbol{r}^* \mathbf{1}^T)^T,$$

where we have a scaling factor $1 - \mu$ and a shift μ on the eigenvalues of the original gradient (where $\mu = 0$). Therefore, if c_0 is the rate of convergence for the Euclidean norm with $\mu = 0$, then the rate of

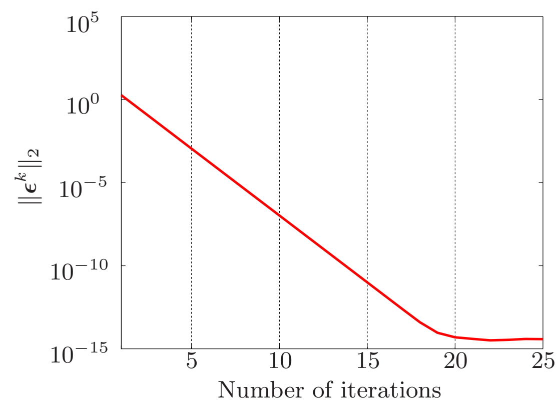


Figure 7.6: The rate of convergence of our method on a real data set.

convergence c_μ for the same norm and a given μ is given by $c_\mu = \mu + (1-\mu)c_0$.

Let us remark that for a singular matrix X, the rate of convergence will be faster. In particular, when X is a rank 1 matrix, we have $X = \boldsymbol{r}^*\mathbf{1}^T$ (every object receives p identical votes from the raters) and our method converges in one step.

The condition on k. In Propositions 9 and Theorem 8, 11, we assumed the condition

$$k \in \mathcal{K}$$

to guarantee uniqueness or convergence. But, there exist greater values of k such that the minimum of E remains unique and the previous methods converge to this minimum. We let as a conjecture that the systems (7.9,7.10) and (7.26,7.27) converge to a unique minimum when we replace the hypercube $\mathcal{H}$ by the convex hull $\mathcal{P}$ in Eq. (7.6) in the definition of the set $\mathcal{K}$ in Eq. (7.20).

By increasing k such that $k \geq \sup \mathcal{K}$, we allow the maxima of E to appear in $\mathcal{H}$, see Fig. 7.3(c). Then, we need to verify during the iteration steps if $(\boldsymbol{r}^t)$ remains in the basin of attraction of E. However if the sequence $(\mathbf{1}^T\boldsymbol{w}^t)$ remains positive, the arguments in proof of Theorem 8 remain valid and the sequence $(E(\boldsymbol{r}^t))$ is monotonically decreasing. It eventually converges to the minimum of E in $\mathcal{H}$ provided that saddle points and maxima are avoided. This can be achieved by using the Hessian of the energy function E and to check if all its eigenvalues are positive at a stationary point. If not, we take any direction of descent to continue the iterations.

The idea of increasing k is to make the discriminant function g more penalizing and therefore to have a better separation between honest and dishonest raters. A possibility is to take during the first steps the largest value of k such that the weights remain nonnegative, that is at time t,

$$k^t : \min_j \boldsymbol{w}_j^t = 0, \tag{7.29}$$

where $\boldsymbol{w}^t$ is now given by

$$G(\boldsymbol{r}^t, k^t) = \mathbf{1} - k^t \, \frac{1}{n} \begin{pmatrix} \|\boldsymbol{x}_1 - \boldsymbol{r}^t\|_2^2 \\ \vdots \\ \|\boldsymbol{x}_p - \boldsymbol{r}^t\|_2^2 \end{pmatrix}.$$

In that manner, the worst rater at time t receives no weight for his own votes and $\boldsymbol{w}^t$ is nonnegative. Usually, the sequence (k^t) converges and one reaches the unique stable fixed point. This will be illustrated later on by simulations.

Operation on the voting matrix. In the definition of the voting matrix, we impose that its entries belong to the interval $[a, b]$. Let X be this initial voting matrix. The two operations of translation and scaling on the voting matrix imply an identical operation on the sequence $(\boldsymbol{r}^t)$ in our method. In the two cases, the proofs only require a simple substitution argument.

Translation. Let $(\tilde{\boldsymbol{r}}^t)$ be the sequence generated by our method with the translated voting matrix

$$\tilde{X} = X + \boldsymbol{s}\mathbf{1}^T.$$

Then, the sequence $(\tilde{\boldsymbol{r}}^t)$ is identical to the sequence $(\boldsymbol{r}^t + \boldsymbol{s})$.

Scaling. If we scale all votes by μ, then another sequence $(\tilde{\boldsymbol{r}}^t)$ is generated with the scaled voting matrix

$$\tilde{X} = \mu X,$$

and the parameters $\tilde{k} = k/\mu^2$ in the discriminant function. Then, we have that the sequence $(\tilde{\boldsymbol{r}}^t)$ is identical to the sequence $(\mu \, \boldsymbol{r}^t)$.

7.4 The other iterative filtering systems

In this section, we compare our method with the two other choices of discriminant function in Eq. (7.30) and (7.31) leading to different quadratic IF systems. But, before that, we briefly discuss the general case of IF systems.

The general case. The definition of an IF system is based on the reputation function F and the filtering function G. Moreover, we could choose another norm for the belief divergence defined in Eq. (7.2). But in that case, for example, Proposition 8 and the existence of an energy function in general is not possible anymore. In general terms, one iteration step of IF systems is defined by

$$\begin{aligned} \boldsymbol{r}^{t+1} &= F(\boldsymbol{w}^t), \\ \boldsymbol{w}^{t+1} &= G(\boldsymbol{r}^{t+1}), \end{aligned}$$

but it does not imply any convergence properties, nor robustness with respect to initial conditions. That system can have several converging solutions and it allows the existence of cycles in the iterative processes.

The discriminant function. We discuss the different properties that quadratic IF systems have depending on their discriminant function g. Then, in the next subsection, we present the candidate functions g in Eq. (7.30,7.31,7.32) and their properties.

Two extreme cases are shown in Fig 7.7: the constant function g_1 leads to equal weights and consequently to merely averaging the votes for the reputations. In that case, the belief divergence is not taken into account for the calculation of the weights and the solution is unique. On the other hand, the function g_4 that gives positive weights only for the raters with zero belief divergence leads to quadratic IF systems with as many reputations as vectors $\boldsymbol{x}_j$ of votes. Therefore, the fixed points correspond to very local opinions where the weight of rater j is maximal while those of other raters is minimal.

Clearly, we are interested in intermediate cases with the following compromise: the discriminant function must be sufficiently discriminating to penalize the outliers, but it has to avoid meaningless local solutions.

The number of stable fixed points is determined by the voting matrix X and the discriminant function g. The former gives the distribution of

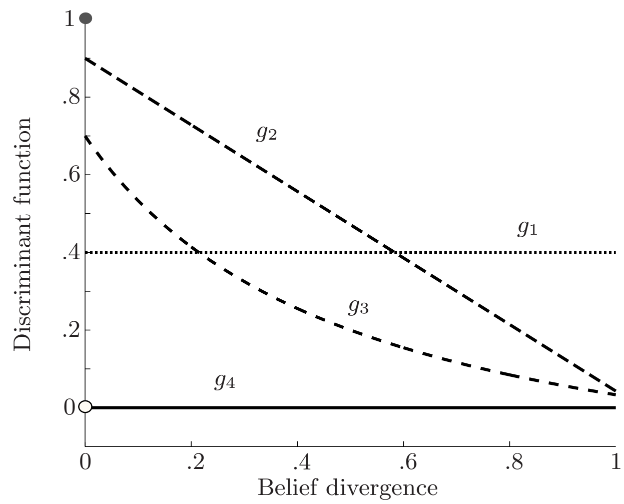


Figure 7.7: (g_1) The discriminant function is constant, weights are independent from the belief divergence, (g_2, g_3) intermediate cases (g_4) weights are null except at the origin.

the votes and it can have clusters of opinions in the data. The latter can be defined such that the cost function always has a unique minimum, or such that several clusters lead to several minima.

When the function g guarantees the uniqueness of the solution, we avoid the choice between different solutions and we also simplify the dynamics of IF systems since multiple stable points implies the existence of unstable points or bifurcations. However, the data may hide two or more different opinions that would be lost in the aggregated solution given by a unique stable fixed point. Then, either the population of raters is separable by extra information (e.g., the age) and we look for a single stable fixed point for each cluster, or the IF system with several stable fixed points is directly interpreted and used as a clustering method where the i^{th} solution gives a weighting vector $\boldsymbol{w}_{(i)}$ and a corresponding reputation vector $\boldsymbol{r}_{(i)}$ coherent with that cluster.

Let us remark that dynamical IF systems with possibly several minimizers mean that we need to track several trajectories during the iterations. Every fixed point will have a basin of attraction depending on the iteration function. We will then need to provide methods that avoid reproducing identical solutions.

The candidate functions. Let us remind the three choices of function g having interesting properties

$$
\begin{aligned}
g(d) &= d^{-k}, && (7.30)\\
g(d) &= e^{-k\,d}, && (7.31)\\
g(d) &= 1 - k\,d. && (7.32)
\end{aligned}
$$

All discriminant functions g are positive and decrease with d for positive k and therefore satisfy the IF-property. However, as already discussed, k must be small enough to keep g positive in Eq. (7.32) and hence to avoid negative weights. If the condition of uniqueness has been established for that case in the previous section, it is not the case for the two other functions. When $k = 0$, we recover the averaging method where the weights are equal. For increasing values of k, the functions g becomes more and more penalizing. Let us analyze in more detail these three candidates that we name the inverse, exponential and affine functions.

Inverse function. The first definition $g(d) = d^{-k}$ was proposed by Laureti et al. in [62, 105] with $k \geq 0$ to apply an iterative filtering similar to the one we present. The choice of function g is based on the maximum of a density function. Assuming that X_{ij} are uncorrelated variables following a multivariate normal distribution centered in $\boldsymbol{r}_i$ with variance $\boldsymbol{\sigma}_j^2$, i.e., $X_{ij} \sim N(\boldsymbol{r}_i, \boldsymbol{\sigma}_j^2)$, then the probability density function f to have the voting matrix X for given $\boldsymbol{r}$ and $\boldsymbol{\sigma}$ is equal to

$$f(X|\boldsymbol{r},\boldsymbol{\sigma}) = \prod_{i,j} \frac{1}{\sqrt{2\pi}\boldsymbol{\sigma}_j} e^{-\frac{(X_{ij}-\boldsymbol{r}_i)^2}{\boldsymbol{\sigma}_j^2}}.$$

Hence, we recover the quadratic IF system with $g(d) = d^{-k}$ with $k = 1$ by considering the following iterations

$$
\begin{aligned}
\boldsymbol{r}^{t+1} &= \arg\max_{\boldsymbol{r}} f(X|\boldsymbol{r},\boldsymbol{\sigma}^t),\\
\boldsymbol{\sigma}^{t+1} &= \sqrt{\boldsymbol{d}^{t+1}},
\end{aligned}
$$

where the square root is applied componentwise on the belief divergence. Therefore, $\boldsymbol{d}_j$ is taken as an estimate of the variance $\boldsymbol{\sigma}_j^2$. The choice $k = .5$ was also proposed in [62, 105] as a variant. Let us remark that in both cases, more than one stable fixed point may appear (see example in next subsection) and that the function g is not defined for $d = 0$. Moreover, unstable fixed points make the iterations sensitive to initial conditions. However, such cases of multiple fixed point is rarely observed when the votes follow a gaussian or an uniform distribution. That is the presence of clusters among the raters that makes more probable two or more fixed points.

Exponential function. The function $g(d) = e^{-k\,d}$ leads to another quadratic IF system based on a similar argument. The probability density function f to have the votes $\boldsymbol{x}_j$ of rater j being given $\boldsymbol{r}$ and the scalar σ is

$$f(\boldsymbol{x}_j|\boldsymbol{r},\sigma) = \prod_i \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(X_{ij}-r_i)^2}{\sigma^2}}. \tag{7.33}$$

This time, we assume that the raters have the same variance σ for their votes. Then we recover the quadratic IF system with $g(d) = e^{-k\,d}$ by considering the iterations on the reputations $\boldsymbol{r}^{t+1} = X\frac{\boldsymbol{w}^t}{\mathbf{1}^T\boldsymbol{w}^t}$ and the ones on the weights

$$\boldsymbol{w}^{t+1} = \begin{bmatrix} f(\boldsymbol{x}_1|\boldsymbol{r}^{t+1},\sigma) \\ \vdots \\ f(\boldsymbol{x}_p|\boldsymbol{r}^{t+1},\sigma) \end{bmatrix} = \left(\sqrt{2\pi}\sigma\right)^{-m} \begin{bmatrix} e^{-\frac{1}{\sigma^2}\boldsymbol{d}_1^{t+1}} \\ \vdots \\ e^{-\frac{1}{\sigma^2}\boldsymbol{d}_p^{t+1}} \end{bmatrix}.$$

The parameter k is then given by the inverse of the variance, that is $1/\sigma^2$. Therefore, by assuming a large variance for all votes, we decrease k and this naturally implies a larger acceptation of divergent opinions making the function g less discriminating.

Affine function. The function g in Eq. (7.32) (leading to our method) links the belief divergence $\boldsymbol{d}$ to the weights $\boldsymbol{w}$: by an affine function. It has the advantage to make the analysis tractable with a direct condition of uniqueness on the parameter k.
Its associated quadratic IF system has also a statistic interpretation. For this purpose, we use the log-likelihood of the density function in Eq.(7.33) for the iterations of the weights. This gives a degree of belief for each rater that was used in [106] but without iterative process. The quadratic IF system with $g(d) = 1 - k\,d$ is recovered by considering the

iterations on the reputations $\boldsymbol{r}^{t+1} = X\frac{\boldsymbol{w}^t}{\mathbf{1}^T\boldsymbol{w}^t}$ and the ones on the weights

$$\boldsymbol{w}^{t+1} = \log \begin{bmatrix} f(\boldsymbol{x}_1|\boldsymbol{r}^{t+1},\sigma) \\ \ldots \\ f(\boldsymbol{x}_p|\boldsymbol{r}^{t+1},\sigma) \end{bmatrix},$$

where log is applied componentwise, and parameter k is given by $\frac{2}{\sigma^2}\log\frac{1}{2\pi\sigma^2}$. The same remark as before can be made about the link between the assumed variance of the votes and the discriminant parameter k.

Let us give the equivalent energy functions for the discriminant functions in Eq. (7.30-7.32) where we use $\boldsymbol{d}_j = \|\boldsymbol{x}_j - \boldsymbol{r}\|_2^2$ and $\boldsymbol{w}_j = g(\boldsymbol{d}_j)$ as two functions of $\boldsymbol{r}$.

function g	$\boldsymbol{r}^*$ minimizes	$\boldsymbol{r}^*$ maximizes
$g(d) = \frac{1}{d}$	$\sum_j \log \boldsymbol{d}_j$	$\prod_j \boldsymbol{w}_j$
$g(d) = \frac{1}{\sqrt{d}}$	$\sum_j -\sqrt{\boldsymbol{d}_j}$	$\sum_j \boldsymbol{w}_j^{-1}$
$g(d) = e^{-k\,d}$	$\sum_j -e^{-k\,\boldsymbol{d}_j}$	$\sum_j \boldsymbol{w}_j$
$g(d) = 1 - k\,d$	$\sum_j \boldsymbol{d}_j - \frac{k}{2}\sum_j \boldsymbol{d}_j^2$	$\sum_j \boldsymbol{w}_j^2$

Let us remark that the function $g(d) = 1 - k\,d$ implies that the system minimizes $\sum_j \boldsymbol{d}_j - \frac{k}{2}\sum_j \boldsymbol{d}_j^2$. The first term $\sum_j \boldsymbol{d}_j$ is minimized by taking the average votes for $\boldsymbol{r}$ and the second term $-\frac{k}{2}\sum_j \boldsymbol{d}_j^2$ is minimized by taking $\boldsymbol{r}$ in $[a,b]^n$. Therefore we have a compromise between the simple average and a solution on the boundary of $\mathcal{H}$ that diverges from this average. The parameter k strengthens the impact of belief divergences on the weights and it makes the solution moving away from the average.

The list of possible discriminant functions can be long. Among other things, we can also recover a form of the Expected Maximization (EM) algorithm. This is obtained when we assume two different distributions for the vector of votes $\boldsymbol{x}_j$ of each rater: (1) with probability α, it follows a multivariate normal distribution with means $\boldsymbol{r}$ and equal variances σ^2, or (2) with probability $1-\alpha$, it follows a multivariate uniform distribution on $[a,b]^n$. Therefore, the first probability density function $f_1(\boldsymbol{x}_j|\boldsymbol{r},\sigma)$ is identical to the one given in Eq.(7.33) and the second probability density function is simply $f_2(\boldsymbol{x}_j) = 1$.

Hence the EM algorithm proceeds in two steps. First we apply the Expectation step. We calculate the probability $\boldsymbol{w}_j^t$ that the votes $\boldsymbol{x}_j$

follows f_1 with the vector of means $\boldsymbol{r}^t$:

$$\boldsymbol{w}_j^t = \frac{\alpha f_1(\boldsymbol{x}_j|\boldsymbol{r}^t, \sigma)}{\alpha f_1(\boldsymbol{x}_j|\boldsymbol{r}^t, \sigma) + (1-\alpha) f_2(\boldsymbol{x}_j)} = \frac{e^{-k\, \boldsymbol{d}_j^t}}{\alpha e^{-k\, \boldsymbol{d}_j^t} + \beta}$$

for $j = 1, \cdots, p$, and with $\beta = (1-\alpha)(2\pi\sigma^2)^{n/2}$ and $k = 1/\sigma^2$. Then follows the Maximization step: we update the vector of means: $\boldsymbol{r}^{t+1} = X \frac{\boldsymbol{w}^t}{\mathbf{1}^T \boldsymbol{w}^t}$. We see that these iterations correspond to a quadratic IF system with the discriminant function

$$g(d) = \frac{1}{\alpha + \beta\, e^{k\, d}}.$$

The convergence properties of the EM algorithm are discussed in [104] and in the example of the next subsection, the sequence $(\boldsymbol{r}^t)$ converges. However, the limit point may be a saddle point and we can have different limit points according to the initial conditions.

Comparison between the three functions. In order to illustrate some of the quadratic IF systems presented above with different discriminant function g, we show the sequences $(\boldsymbol{r}^t)$ and $(\boldsymbol{w}^t)$ for a given voting matrix X and their limit points $\boldsymbol{r}^*$ and $\boldsymbol{w}^*$. We propose the same fictive voting matrix than in the toy example of Section 7.2:

$$X = \begin{bmatrix} 3.3 & 3.4 & 4.9 \\ 4.2 & 4.5 & 2.8 \end{bmatrix},$$

where the two rows correspond to the votes given to two ice skaters and the three columns represent the three raters. As remarked earlier, the third rater tries to favor the first ice skater. If equal weights are given to the raters, we obtain the average votes: 3.87 for the first skater and 3.83 for the second one. We show the different results according to the choice of discriminant functions discussed before.

The iterations using Eq.(7.32) with $k = 1/5$, that is $g(d) = 1 - d/5$, are already given in the toy example of Section 7.2. The following table shows the weights and the reputations for several discriminant functions g and the initial vector $\boldsymbol{w}^0 = \mathbf{1}$.

$g(d)$	$(\boldsymbol{w}^*)^T$	$(\boldsymbol{r}^*)^T$
$1-\frac{1}{5}\,d$	$\begin{bmatrix} 1.0 & 1.0 & 0.8 \end{bmatrix}$	$\begin{bmatrix} 3.8 & 3.9 \end{bmatrix}$
$1-\frac{1}{3}\,d$	$\begin{bmatrix} 1.0 & 1.0 & 0.5 \end{bmatrix}$	$\begin{bmatrix} 3.7 & 4.0 \end{bmatrix}$
d^{-1}	$\begin{bmatrix} 1 & 0 & 0 \end{bmatrix}$	$\begin{bmatrix} 3.3 & 4.2 \end{bmatrix}$
d^{-1} $(*)$	$\begin{bmatrix} 0 & 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 4.9 & 2.8 \end{bmatrix}$
$d^{-1/2}$	$\begin{bmatrix} 35.2 & 4.8 & .7 \end{bmatrix}$	$\begin{bmatrix} 3.3 & 4.2 \end{bmatrix}$
e^{-d}	$\begin{bmatrix} 1.0 & 1.0 & 0.1 \end{bmatrix}$	$\begin{bmatrix} 3.4 & 4.2 \end{bmatrix}$

Rows 1-2. The first row is the same discriminant function as before and the second one considers another k which is more penalizing. Consequently, the third rater is more severely weighted and larger k would continue to decrease his weight.

Rows 3-4. The iterations are numerically unstable because the belief divergence $\boldsymbol{d}_1$ of rater 1 tends to zero, making his weight infinite. Disregarding this instability, the normalized weights tend to $[1\,0\,0]^T$ and therefore the convergent reputations vector corresponds to the first column of X. However the final reputations and weights depend on the initial vector $\boldsymbol{w}^0$. The fourth row $(*)$ considers $\boldsymbol{w}^0 = [1\ 1\ 4]^T$ instead of equal weights. Then the third rater gains all the confidence and the final reputations are given by his votes: the third column of X.

Row 5. In this case, the initial point does not matter, provided that we avoid a zero belief divergence. We see that this function almost disqualifies the third rater: it makes a large difference between the first two raters and it is more penalizing than $g(d) = 1 - \frac{1}{3}\,d$.

Row 6. Finally, we look at the iterations given by Eq.(7.31): $g(d) = e^{-k\,d}$. From a certain k, several final reputations are possible depending on the initial vector $\boldsymbol{w}^0$. Similarly as for $g(d) = d^{-1}$, if $\boldsymbol{w}^0$ favors the third rater, then this rater will have an advantageous weight at the end. For $k = 1$, we avoid multiple reputations and we obtain the results in row 6.

7.5 Sparsity pattern and dynamical votes

This section extends the convergence properties of our method (the system (7.9,7.10) to the case where the voting matrix has some sparsity pattern, that is when an object is not evaluated by all raters. Moreover we analyze dynamical voting matrices representing votes that evolve over time.

Sparsity pattern. In general, the structure of real data is sparse. We hardly find a set of raters and objects with a vote for all possible pairs. An absence of a vote for object i from rater j will imply that the entry (i, j) of the matrix X is equal to zero, that is, by using the adjacency matrix A,

$$\text{if } A_{ij} = 0, \text{ then } X_{ij} = 0.$$

These entries must not be considered as votes but instead as missing values. Therefore the previous equations presented in matrix form require some modifications that will include the adjacency matrix A. We write the new equations and their implications using the order of the previous section.

The belief divergence for IF systems in Eq.(7.2) becomes

$$\boldsymbol{d} = \begin{pmatrix} \frac{1}{\boldsymbol{n}_1}\|\boldsymbol{x}_1 - \boldsymbol{a}_1 \circ \boldsymbol{r}\|_2^2 \\ \vdots \\ \frac{1}{\boldsymbol{n}_p}\|\boldsymbol{x}_p - \boldsymbol{a}_p \circ \boldsymbol{r}\|_2^2 \end{pmatrix}. \tag{7.34}$$

where $\boldsymbol{a}_j$ is the j^{th} column of the adjacency matrix A and $\boldsymbol{n}_j$ is the j^{th} entry of the vector $\boldsymbol{n}$ containing the numbers of votes given to each item, i.e.,

$$\boldsymbol{n} = A^T\mathbf{1}.$$

If A is dense, i.e., $A = \mathbf{1}\mathbf{1}^T$, then we recover the previous equations with $\boldsymbol{n} = n\mathbf{1}$ where n is the total number of objects.

The reputation function F of quadratic IF systems remains the weighted average of the votes, and is given in matrix form by

$$F(\boldsymbol{w}) = \frac{[X\boldsymbol{w}]}{[A\boldsymbol{w}]},$$

where $\frac{[\cdot]}{[\cdot]}$ is the componentwise division. Let us remark that every entry of $A\boldsymbol{w}$ must be strictly positive. This means that every object is evaluated by at least one rater with nonzero weight.

With these modifications, the iterations and the fixed point of quadratic IF systems are given by quadratic equations in $\boldsymbol{r}$ and $\boldsymbol{w}$

$$\begin{aligned} (A \circ \boldsymbol{r}^{t+1}\mathbf{1}^T)\,\boldsymbol{w}^t &= X\boldsymbol{w}^t && (7.35)\\ (A \circ \boldsymbol{r}^{*}\mathbf{1}^T)\,\boldsymbol{w}^* &= X\boldsymbol{w}^*. && (7.36) \end{aligned}$$

where $\boldsymbol{w}^* = G(\boldsymbol{r}^*)$. Hence, the fixed points in Eq. (7.36) amount to the roots of the function

$$D(\boldsymbol{r}) = \frac{2}{n}(A \circ \boldsymbol{r}\mathbf{1}^T - X) \cdot G(\boldsymbol{r}),$$

that is the sparse version of Eq. (7.13). Similarly, we show that it is the gradient of some energy function introduced in the next proposition generalizing Proposition 8.

Proposition 13. *The fixed points of quadratic IF systems with integrable discriminant function g, are the singular points of the energy function*

$$E(\boldsymbol{r}) = \frac{1}{n}\sum_{j=1}^{p} \boldsymbol{n}_j \int_0^{\boldsymbol{d}_j(\boldsymbol{r})} g(u)\,du + c, \qquad (7.37)$$

where $\boldsymbol{d}_j$ is the belief divergence of rater j that depends on $\boldsymbol{r}$ and $c \in \mathbb{R}$ is a constant. Moreover one iteration step in quadratic IF systems corresponds to a dilated steepest descent direction with a particular step size

$$\boldsymbol{r}^{t+1} = \boldsymbol{r}^t - \boldsymbol{\alpha}^t \circ \nabla_{\boldsymbol{r}} E(\boldsymbol{r}^t) \qquad (7.38)$$

with $\boldsymbol{\alpha}^t = \frac{n}{2}\frac{[\mathbf{1}]}{[A\boldsymbol{w}^t]}$.

Proof: We have $\nabla_{\boldsymbol{r}} E(\boldsymbol{r}) = \nabla_{\boldsymbol{r}} \boldsymbol{d}^T \cdot \nabla_{\boldsymbol{d}} E(\boldsymbol{r})$ with

$$\begin{aligned} \nabla_{\boldsymbol{r}} \boldsymbol{d}^T &= -2\,\frac{\left[X - A \circ \boldsymbol{r}\mathbf{1}^T\right]}{[\mathbf{1}\boldsymbol{n}^T]} \\ \nabla_{\boldsymbol{d}} E(\boldsymbol{r}) &= \frac{1}{n}\left[\boldsymbol{n} \circ G(\boldsymbol{r})\right], \end{aligned}$$

Therefore a stationary point $\boldsymbol{r}^*$ in E satisfies

$$\begin{aligned} -\frac{2}{n}\,\frac{\left[X - A\circ \boldsymbol{r}^*\boldsymbol{1}^T\right]}{\left[\boldsymbol{1}\boldsymbol{n}^T\right]}\Big(\boldsymbol{n}\circ G(\boldsymbol{r}^*)\Big) &= 0 \\ -\frac{2}{n}\left(X - A\circ \boldsymbol{r}^*\boldsymbol{1}^T\right)G(\boldsymbol{r}^*) &= 0 \\ \left(A\circ \boldsymbol{r}^*\boldsymbol{1}^T\right)\boldsymbol{w}^* &= X\boldsymbol{w}^*, \end{aligned}$$

which corresponds to the fixed point equation given in Eq.(7.36).

We also have $\nabla_{\boldsymbol{r}} E(\boldsymbol{r}^t) = \nabla_{\boldsymbol{r}}(\boldsymbol{d}(\boldsymbol{r}^t))^T \cdot \nabla_{\boldsymbol{d}} E(\boldsymbol{r}^t)$ with

$$\begin{aligned} \nabla_{\boldsymbol{r}}(\boldsymbol{d}(\boldsymbol{r}^t))^T &= -2\,\frac{\left[X - A\circ \boldsymbol{r}^t\boldsymbol{1}^T\right]}{\left[\boldsymbol{1}\boldsymbol{n}^T\right]} \\ \nabla_{\boldsymbol{d}} E(\boldsymbol{r}^t) &= \frac{1}{n}\left[\boldsymbol{n}\circ G(\boldsymbol{r}^t)\right] = \frac{1}{n}\left[\boldsymbol{n}\circ \boldsymbol{w}^t\right]. \end{aligned}$$

Therefore

$$\begin{aligned} \nabla_{\boldsymbol{r}} E(\boldsymbol{r}^t) &= -\frac{2}{n}(X - A\circ \boldsymbol{r}^t\boldsymbol{1}^T)\boldsymbol{w}^t \qquad (7.39) \\ &= -\frac{2}{n}(A\boldsymbol{w}^t)\circ(\boldsymbol{r}^{t+1} - \boldsymbol{r}^t) \\ &= -\frac{\left[\boldsymbol{r}^{t+1} - \boldsymbol{r}^t\right]}{\left[\boldsymbol{\alpha}^t\right]}. \end{aligned}$$

■

The number of votes $\boldsymbol{n}_j$ gives somehow a weight of importance for the minimization of the surface $\int_0^{\boldsymbol{d}_j} g(u)\,du$. Therefore a rater with more votes receives more attention in the minimization process. Moreover the table in Section 7.4 becomes

function g	$\boldsymbol{r}^*$ minimizes	$\boldsymbol{r}^*$ maximizes
$g(d) = \frac{1}{d}$	$\sum_j \boldsymbol{n}_j \log \boldsymbol{d}_j$	$\prod_j \boldsymbol{w}_j^{\boldsymbol{n}_j}$
$g(d) = \frac{1}{\sqrt{d}}$	$\sum_j -\boldsymbol{n}_j\sqrt{\boldsymbol{d}_j}$	$\sum_j \boldsymbol{n}_j \boldsymbol{w}_j^{-1}$
$g(d) = e^{-k\,d}$	$\sum_j -\boldsymbol{n}_j e^{-k\,\boldsymbol{d}_j}$	$\sum_j \boldsymbol{n}_j \boldsymbol{w}_j$
$g(d) = 1 - k\,d$	$\sum_j \boldsymbol{n}_j\boldsymbol{d}_j - \boldsymbol{n}_j\frac{k}{2}\sum_j \boldsymbol{d}_j^2$	$\sum_j \boldsymbol{n}_j \boldsymbol{w}_j^2$

Then the iteration steps of our method for sparse matrix X are given by

$$\boldsymbol{r}^{t+1} = \frac{\lceil X\boldsymbol{w}^t \rceil}{[A\boldsymbol{w}^t]}, \tag{7.40}$$

$$\boldsymbol{w}^{t+1} = \mathbf{1} - k \begin{pmatrix} \frac{1}{\boldsymbol{n}_1}\|\boldsymbol{x}_1 - \boldsymbol{a}_1 \circ \boldsymbol{r}^{t+1}\|_2^2 \\ \vdots \\ \frac{1}{\boldsymbol{n}_p}\|\boldsymbol{x}_p - \boldsymbol{a}_p \circ \boldsymbol{r}^{t+1}\|_2^2 \end{pmatrix}, \tag{7.41}$$

where the sequence $(\boldsymbol{r}^t)$ still remains in $\mathcal{H} = [a, b]^n$ for nonnegative weights $\boldsymbol{w}^t$, $t \geq 0$. Then its associated energy function is similar than the previous one given in Eq. (7.18):

$$E(\boldsymbol{r}) = -\frac{1}{2\,kn}\,\boldsymbol{w}^T[\boldsymbol{w} \circ \boldsymbol{n}], \tag{7.42}$$

where $\boldsymbol{w}$ depends on $\boldsymbol{r}$ according to the function $G(\boldsymbol{r})$. Therefore E is a fourth-order polynomial,

$$E(\boldsymbol{r}) = -\frac{1}{2\,k}\sum_{j=1}^{p}\frac{\boldsymbol{n}_j}{n}\left(1 - k\,\frac{1}{n}\|\boldsymbol{x}_j - \boldsymbol{a}_j \circ \boldsymbol{r}\|_2^2\right)^2.$$

Hence, Proposition 9 remains valid and the arguments are similar. The only difference is in the definition of the set $\mathcal{K}$ in Eq. (7.20) which is now given by

$$\mathcal{K} = \{k \in \mathbb{R}_{\geq 0} : \mathbf{1} - k \begin{pmatrix} \frac{1}{\boldsymbol{n}_1}\|\boldsymbol{x}_1 - \boldsymbol{a}_1 \circ \boldsymbol{r}\|_2^2 \\ \vdots \\ \frac{1}{\boldsymbol{n}_p}\|\boldsymbol{x}_p - \boldsymbol{a}_p \circ \boldsymbol{r}\|_2^2 \end{pmatrix} > 0 \text{ for all } \boldsymbol{r} \in \mathcal{H}\}.$$

Therefore the condition on k to guarantee positive weights becomes $k \in \mathcal{K}$ according to that new definition.

Our method and its coordinate version for sparse matrix X still converge with the property that the sequence $(E(\boldsymbol{r}^t))$ decreases. The proofs are closely related to the ones presented in Theorem 8 and 11. We give only the proof for our method since the one for the coordinate version is very similar.

Proposition 14. *If $k \in \mathcal{K}$, the system (7.40,7.41) converges to the unique fixed point $\boldsymbol{r}^* \in \mathcal{H}$.*

Proof: First, we show that the energy function E decreases between any two iterations, i.e., $E(\boldsymbol{r}^{t+1}) \leq E(\boldsymbol{r}^t)$ for all $t \geq 0$. By Eq.(7.18), this is equivalent to prove that $(\boldsymbol{w}^{t+1})^T(\boldsymbol{w}^{t+1} \circ \boldsymbol{n}) \geq (\boldsymbol{w}^t)^T(\boldsymbol{w}^t \circ \boldsymbol{n})$. Let us express $\boldsymbol{w}^{t+1}$ in terms of $\boldsymbol{w}^t$, then we obtain

$$\begin{aligned}
\boldsymbol{w}^{t+1} &= \mathbf{1} - \frac{[k\,\mathbf{1}]}{[\boldsymbol{n}]} \circ [X^T - A^T \circ \mathbf{1}(\boldsymbol{r}^{t+1})^T]^{\circ 2}\mathbf{1} \\
&= \mathbf{1} - \frac{[k\,\mathbf{1}]}{[\boldsymbol{n}]} \circ [X^T - A^T \circ \mathbf{1}(\boldsymbol{r}^t)^T - A^T \circ \mathbf{1}(\boldsymbol{r}^{t+1} - \boldsymbol{r}^t)^T]^{\circ 2}\mathbf{1} \\
&\qquad \text{(by Lemma 8 with } M^T = X^T - A^T \circ \mathbf{1}(\boldsymbol{r}^t)^T \text{ and } \boldsymbol{c} = \boldsymbol{r}^{t+1} - \boldsymbol{r}^t,) \\
&= \boldsymbol{w}^t + \frac{[k\,\mathbf{1}]}{[\boldsymbol{n}]} \circ \big(2\,(X^T - A^T \circ \mathbf{1}(\boldsymbol{r}^t)^T)(\boldsymbol{r}^{t+1} - \boldsymbol{r}^t) - A^T(\boldsymbol{r}^{t+1} - \boldsymbol{r}^t)^{\circ 2}\big) \\
&= \boldsymbol{w}^t + \frac{[k\,\mathbf{1}]}{[\boldsymbol{n}]} \circ \boldsymbol{q}.
\end{aligned}$$

where $\boldsymbol{q} := \big(2\,(X^T - A^T \circ \mathbf{1}(\boldsymbol{r}^t)^T)(\boldsymbol{r}^{t+1} - \boldsymbol{r}^t) - A^T(\boldsymbol{r}^{t+1} - \boldsymbol{r}^t)^{\circ 2}\big)$. Hence,

$$\begin{aligned}
(\boldsymbol{w}^{t+1})^T(\boldsymbol{w}^{t+1} \circ \boldsymbol{n}) &= (\boldsymbol{w}^t + \frac{[k\,\mathbf{1}]}{[\boldsymbol{n}]} \circ \boldsymbol{q})^T(\boldsymbol{w}^t \circ \boldsymbol{n} + k\,\boldsymbol{q}) \\
&= (\boldsymbol{w}^t)^T(\boldsymbol{w}^t \circ \boldsymbol{n}) + (\frac{[k^2\mathbf{1}]}{[\boldsymbol{n}]} \circ \boldsymbol{q})^T\boldsymbol{q} + 2k\,\boldsymbol{q}^T\boldsymbol{w}^t.
\end{aligned}$$

Therefore, it is sufficient to show that $\boldsymbol{q}^T\boldsymbol{w}^t \geq 0$:

$$\begin{aligned}
\boldsymbol{q}^T\boldsymbol{w}^t &= 2\,(\boldsymbol{r}^{t+1} - \boldsymbol{r}^t)^T(X - A \circ \boldsymbol{r}^t\mathbf{1}^T)\boldsymbol{w}^t - ((\boldsymbol{r}^{t+1} - \boldsymbol{r}^t)^{\circ 2})^T A\boldsymbol{w}^t \\
&= 2\,((\boldsymbol{r}^{t+1} - \boldsymbol{r}^t)^{\circ 2})^T A\boldsymbol{w}^t - ((\boldsymbol{r}^{t+1} - \boldsymbol{r}^t)^{\circ 2})^T A\boldsymbol{w}^t \\
&= ((\boldsymbol{r}^{t+1} - \boldsymbol{r}^t)^{\circ 2})^T A\boldsymbol{w}^t,
\end{aligned}$$

and since every entry of $A\boldsymbol{w}^t$ is larger than some $\delta > 0$ by the condition on k, the energy is strictly decreasing when $\boldsymbol{r}^{t+1} \neq \boldsymbol{r}^t$:

$$E(\boldsymbol{r}^{t+1}) - E(\boldsymbol{r}^t) \leq -\frac{\delta}{n}\,\|\boldsymbol{r}^{t+1} - \boldsymbol{r}^t\|_2^2, \tag{7.43}$$

as in addition E is lower bounded in $\mathcal{H}$, the sequence $(\boldsymbol{r}^t)$ converges to a single limit point $\boldsymbol{r}^* \in \mathcal{H}$. Then the gradient of E must be zero in $\boldsymbol{r}^*$ because we have by Eq.(7.15)

$$\begin{aligned}
\|\nabla_{\boldsymbol{r}}E(\boldsymbol{r}^t)\|_2 &= \frac{2}{n}\,\big\|A\boldsymbol{w}^t \circ (\boldsymbol{r}^{t+1} - \boldsymbol{r}^t)\big\|_2 \\
&\leq \frac{2}{n}\|\boldsymbol{r}^{t+1} - \boldsymbol{r}^t\|_2,
\end{aligned}$$

where we used that $A\boldsymbol{w}^t$ is componentwise upper bounded by $\mathbf{1}$. Therefore the gradient is zero at $\boldsymbol{r}^*$ and by Proposition 9 (also valid for our method with sparse matrix), the point $\boldsymbol{r}^*$ is the unique fixed point. ∎

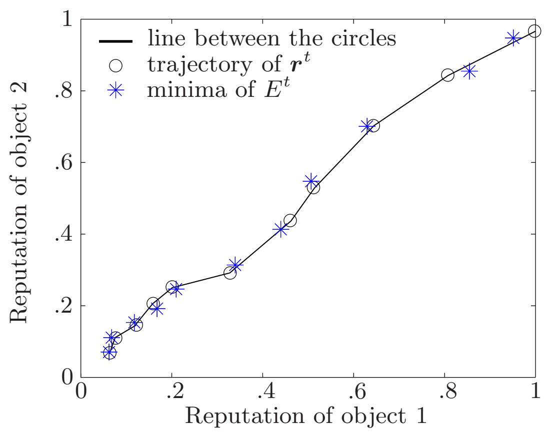


Figure 7.8: Trajectory of reputations (the circles) given by system in Eq. (7.44,7.45) with some dynamical voting matrix and the evolving parameter k^t defined in Eq.(7.29). Trajectory of the minima of the energy function E^t (the stars). The difference between circles and stars increases when votes change faster, this also corresponds to larger steps. For fixed values of k, those differences are generally smaller.

Finally, the rate of convergence can be proved to be q-linear using similar arguments as in Proposition 12. Moreover, the discussion on the condition $k \in \mathcal{K}$ remains valid and it is possible to take larger k in order to better separate honest raters from spammers. In particular, the choice of the largest k expressed in Eq. (7.29) at the end of Section 7.4 remains an interesting variant provided that we avoid stationary points before reaching the minimum.

We remark that the earlier analysis can still be applied when we introduce a sparsity pattern in the voting matrix.

Dynamical votes. We consider in this section the case of time-varying votes. Formally, we have discrete sequences

$$(X^t)_{t\geq 1}, \qquad (A^t)_{t\geq 1}$$

of voting matrices and adjacency matrices evolving over time t. Hence our method in Eq. (7.40,7.41) for dynamical (and sparse) voting matrices is then given by

$$\boldsymbol{r}^{t+1} = F_{t+1}(\boldsymbol{w}^t) = \frac{[X^{t+1}\boldsymbol{w}^t]}{[A^{t+1}\boldsymbol{w}^t]}, \tag{7.44}$$

$$\boldsymbol{w}^{t+1} = G_{t+1}(\boldsymbol{r}^{t+1}) = \mathbf{1} - k \begin{pmatrix} \frac{1}{\boldsymbol{n}_1^{t+1}} \|\boldsymbol{x}_1^{t+1} - \boldsymbol{a}_1^{t+1} \circ \boldsymbol{r}^{t+1}\|_2^2 \\ \vdots \\ \frac{1}{\boldsymbol{n}_p^{t+1}} \|\boldsymbol{x}_p^{t+1} - \boldsymbol{a}_p^{t+1} \circ \boldsymbol{r}^{t+1}\|_2^2 \end{pmatrix}. \tag{7.45}$$

We already now that for subsequent constant matrices X^t with $T_1 \le t \le T_2$, the iterations on $\boldsymbol{r}^t$ and $\boldsymbol{w}^t$ of system (7.44,7.45) and its variant for coordinate descent tend to fixed vectors $\boldsymbol{r}^*$ and $\boldsymbol{w}^*$ provided that $k \in \mathcal{K}$.

In fact, each iteration on $\boldsymbol{r}^t$ decreases an energy function E^t dependent on the time t. Intuitively, if the votes change hardly over time, then the sequence of minima of E^t evolves slowly. Therefore different initial conditions will eventually converge to the same trajectory that follows this sequence of minima.

If we have significant changes in the votes, reflecting a lack of consensus in the opinions, then no such convergence result can be established. However, in the specific case of 2−periodic sequences of voting matrices, we have a 2-periodic solution, that is, $\lim_{\boldsymbol{t}\to\infty} \boldsymbol{r}^{2t+1} = \boldsymbol{r}^{1*}$ and $\lim_{\boldsymbol{t}\to\infty} \boldsymbol{r}^{2t} = \boldsymbol{r}^{2*}$.

Proposition 15. *Let the matrices X, A be such that*

$$\begin{aligned} X^{2i+1} &= X^1, & X^{2i} &= X^2 \\ A^{2i+1} &= A^1, & A^{2i} &= A^2 \end{aligned}$$

for $i \in \mathbb{N}$, and a constant number of votes, i.e.,

$$(A^1)^T \mathbf{1} = \boldsymbol{n}^1 = \boldsymbol{n}^2 = (A^2)^T \mathbf{1}.$$

The system (7.44,7.45) and its variant for coordinate descent converge to a unique 2-periodic solution $(\boldsymbol{r}^{1*}, \boldsymbol{r}^{2*})$.

Proof: The arguments are similar to the ones in Proposition 9 to prove the uniqueness of the fixed point, and in Proposition 14 for the convergence to this unique fixed point. They only differ by the definition of the set $\mathcal{K}$ that is now given by

$$\mathcal{K} = \{k \in \mathbb{R}_{\ge 0} : G_t(\boldsymbol{r} > 0) \text{ for all } t \ge 1 \text{ and } \boldsymbol{r} \in \mathcal{H})\}$$

and the energy function

$$E(\boldsymbol{r}^1, \boldsymbol{r}^2) = -\frac{1}{2\,kn}\,(\boldsymbol{w}^1)^T(\boldsymbol{w}^2 \circ \boldsymbol{n}^1),$$

where $\boldsymbol{w}^1 = G_1(\boldsymbol{r}^1)$ and $\boldsymbol{w}^2 = G_2(\boldsymbol{r}^2)$. The gradient of E is given by

$$\begin{aligned}
\begin{bmatrix} \nabla_{\boldsymbol{r}^1} E(\boldsymbol{r}^1, \boldsymbol{r}^2) \\ \nabla_{\boldsymbol{r}^2} E(\boldsymbol{r}^1, \boldsymbol{r}^2) \end{bmatrix} &= -\frac{1}{2\,kn} \begin{bmatrix} (\nabla_{\boldsymbol{r}^1} G_1(\boldsymbol{r}^1)^T)(\boldsymbol{w}^2 \circ \boldsymbol{n}^1) \\ (\nabla_{\boldsymbol{r}^2} G_2(\boldsymbol{r}^2)^T)(\boldsymbol{w}^1 \circ \boldsymbol{n}^1) \end{bmatrix} \\
&= -\frac{1}{2\,kn} \begin{bmatrix} -2k \frac{\left[X^1 - A^1 \circ \boldsymbol{r}^1 \boldsymbol{1}^T\right]}{\left[\boldsymbol{1}(\boldsymbol{n}^1)^T\right]} (\boldsymbol{w}^2 \circ \boldsymbol{n}^1) \\ \\ -2k \frac{\left[X^2 - A^2 \circ \boldsymbol{r}^2 \boldsymbol{1}^T\right]}{\left[\boldsymbol{1}(\boldsymbol{n}^2)^T\right]} (\boldsymbol{w}^1 \circ \boldsymbol{n}^1) \end{bmatrix} \\
&= \frac{1}{n} \begin{bmatrix} X^1 \boldsymbol{w}^2 - A^1 \boldsymbol{w}^2 \circ \boldsymbol{r}^1 \\ X^2 \boldsymbol{w}^1 - A^2 \boldsymbol{w}^1 \circ \boldsymbol{r}^2 \end{bmatrix},
\end{aligned}$$

that gradient is equal to zero at the 2-periodic solution $(\boldsymbol{r}^{1*}, \boldsymbol{r}^{2*})$. Because of similar arguments that those ones in Proposition 9, that 2-periodic solution is unique and the energy function decreases at each iteration. Similarly to Proposition 14, we express $\boldsymbol{w}^{t+1}$ in terms of $\boldsymbol{w}^{t-1}$, then we obtain

$$\begin{aligned}
\boldsymbol{w}^{t+1} &= \boldsymbol{1} - \frac{[k\,\boldsymbol{1}]}{[\boldsymbol{n}^1]} \circ [(X^{t+1})^T - (A^{t+1})^T \circ \boldsymbol{1}(\boldsymbol{r}^{t+1})^T]^{\circ 2}\boldsymbol{1} \\
&= \boldsymbol{w}^{t-1} + \frac{[k\,\boldsymbol{1}]}{[\boldsymbol{n}^1]} \circ \Big(2\,((X^{t+1})^T - (A^{t+1})^T \circ \boldsymbol{1}(\boldsymbol{r}^{t-1})^T)(\boldsymbol{r}^{t+1} - \boldsymbol{r}^{t-1}) \\
&\qquad - (A^{t+1})^T(\boldsymbol{r}^{t+1} - \boldsymbol{r}^{t-1})^{\circ 2} \Big) \\
&= \boldsymbol{w}^{t-1} + \frac{[k\,\boldsymbol{1}]}{[\boldsymbol{n}^1]} \circ \boldsymbol{q}, \qquad (7.46)
\end{aligned}$$

$$\boldsymbol{q} := \Big(2\,((X^{t+1})^T - (A^{t+1})^T \circ \boldsymbol{1}(\boldsymbol{r}^{t-1})^T)(\boldsymbol{r}^{t+1} - \boldsymbol{r}^{t-1}) - (A^{t+1})^T(\boldsymbol{r}^{t+1} - \boldsymbol{r}^{t-1})^{\circ 2} \Big).$$

The energy function decreases if and only if

$$(\boldsymbol{w}^t)^T(\boldsymbol{w}^{t+1} \circ \boldsymbol{n}^1) \geq (\boldsymbol{w}^{t-1})^T(\boldsymbol{w}^t \circ \boldsymbol{n}^1).$$

Using (7.46), this last condition becomes $\boldsymbol{q}^T \boldsymbol{w}^t \geq 0$ and the rest of the proof uses the same arguments than for Proposition 14 except that we have $\|\boldsymbol{r}^{t+1} - \boldsymbol{r}^{t-1}\|_2^2$ instead of $\|\boldsymbol{r}^{t+1} - \boldsymbol{r}^t\|_2^2$. ■

Therefore, for 2−periodic voting matrices, any iteration converges to a unique 2−periodic trajectory determined by $(\boldsymbol{r}^{0*}, \boldsymbol{r}^{1*})$ that maximizes again some weighted sum of $\boldsymbol{w}^1$ and $\boldsymbol{w}^2$ depending on the number of votes $\boldsymbol{n}^1$. However, that Proposition is not valid anymore when the number of votes is different from one step to another, i.e., $\boldsymbol{n}^1 \neq \boldsymbol{n}^2$. In that case, the equations of the 2−periodic fixed points can not be anymore the gradient of any energy function that is a fourth-order polynomial. By contradiction, let $E(\boldsymbol{r}^1, \boldsymbol{r}^2)$ be a fourth-order polynomial that has a gradient proportional to

$$\begin{bmatrix} f_1(\boldsymbol{r}^1, \boldsymbol{r}^2) \\ f_2(\boldsymbol{r}^1, \boldsymbol{r}^2) \end{bmatrix} = \begin{bmatrix} \left(X^1 - A^1 \circ (\boldsymbol{r}^1 \mathbf{1}^T)\right) G_2(\boldsymbol{r}^2) \\ \left(X^2 - A^2 \circ (\boldsymbol{r}^2 \mathbf{1}^T)\right) G_1(\boldsymbol{r}^1) \end{bmatrix}.$$

Since the Hessian of E must be symmetric, we should have $\nabla_{\boldsymbol{r}^2} f_1 = (\nabla_{\boldsymbol{r}^1} f_2)^T$ that is equivalent to

$$\nabla_{\boldsymbol{r}^2} G_2(\boldsymbol{r}^2) \left(X^1 - A^1 \circ (\boldsymbol{r}^1 \mathbf{1}^T)\right)^T = \left(X^2 - A^2 \circ (\boldsymbol{r}^2 \mathbf{1}^T)\right) \left(\nabla_{\boldsymbol{r}^1} G_1(\boldsymbol{r}^1)\right)^T,$$

$$\frac{\left[X^2 - A^2 \circ \boldsymbol{r}^2 \mathbf{1}^T\right]}{[\mathbf{1}(\boldsymbol{n}^2)^T]} \left(X^1 - A^1 \circ (\boldsymbol{r}^1 \mathbf{1}^T)\right)^T = \left(X^2 - A^2 \circ (\boldsymbol{r}^2 \mathbf{1}^T)\right) \frac{\left[(X^1 - A^1 \circ \boldsymbol{r}^1 \mathbf{1}^T)^T\right]}{[\boldsymbol{n}^1 \mathbf{1}^T]},$$

that is satisfied only for $\boldsymbol{n}^1 = \boldsymbol{n}^2$. Similarly, for p−periodic sequences of voting matrices, with $p \geq 3$, there is no energy function $E(\boldsymbol{r}^1, \boldsymbol{r}^2, \boldsymbol{r}^3)$ that is a fourth-order polynomial and the uniqueness of the stable p−periodic trajectory is not preserved. Fig. 7.5 shows a trajectory tending to a 5-periodic trajectory, but with another initial condition, it may converge to another 5-periodic trajectory.

7.6 Computer simulations

We illustrate our method with the updates on the parameter k given in Eq.(7.29). Two sets of data are used for this purpose:

1. the votes of 43 countries during the final of the EuroVision 2008;

2. the votes of 943 movie lovers in the website of MovieLens.

We will see how and who our method penalizes through the iterations. In the first set of data, we compare the difference in the ranking used by Eurovision (average of the votes) and the ranking obtained by our method. Then we observe a posteriori who has been penalized by our method and why. The second set of data is used in order to verify the desired property mentioned in the introduction: *raters diverging often from other raters' opinion are less taken into account.* For this

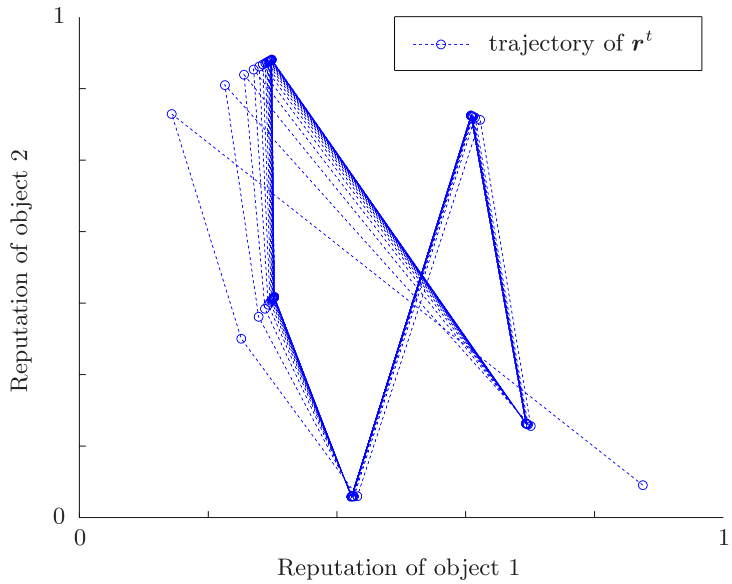


Figure 7.9: Trajectory of reputations (circles) for a 5-periodic voting matrix

purpose, we added fictional raters that diverge from the original raters. Therefore, we have a priori a subset of the raters that is expected to be penalized by our method.

EuroVision. During the final of EuroVision in 2008, 43 countries have evaluated 25 songs. Each song represents a country and no country among the 43 ones may vote for itself. Each country distributes his votes that are taken from $\{0, 1, 2, 3, 4, 5, 6, 7, 8, 10, 12\}$ and only the vote zero can be given several times while the other ones must be given exactly once.

We can expect that votes are driven by linguistic and cultural proximities between countries. However, vote alliances may appear between nearby countries by, for example, exchanging the maximal votes (as discussed in [26]). Such raters should be penalized when they are, in addition, not in agreement with the other votes.

For this example we focus on the final ranking rather than on the reputations themselves. Clearly, the changes in the rankings due to dif-

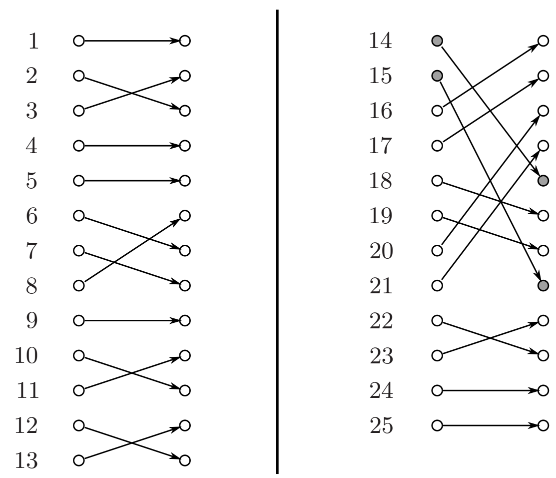


Figure 7.10: Difference of the rankings between the affine IF method and the average of the votes: for example the second and third positions switch and two countries (grey nodes) drop by 4 and 6 positions when we use the affine IF method.

ferent weights for the votes are sensitive and complex, but this final ranking is the main output of the competition and therefore its perturbation is crucial. We then compare in Fig. 7.10 the ranking obtained by taking the average of the votes used by the EuroVision jury and the one given by the affine IF method.

Let us remind that the parameter k plays a role of discrimination between marginal and reasonable raters. For instance, small k do not change the order in the original ranking, and increasing values of k make appear greater jumps in the ranking list. We briefly highlight two phenomena shown in Fig. 7.10: the switch between the second and the third country, and the loss of 6 positions for the 15^{th} in the original ranking.

The switch occurs between Ukraine and Greece that already had a small difference in the original ranking: averages of 5.5 and 5.2. These two countries received opposite votes from many raters. For instance, we find 3 times that one of those countries receives 0 while the other receives the maximum 12. This leads to significant changes when the votes of the raters are weighted. The switch shows that raters favorable to Greece were more objective according to our definition of the belief

divergence.
Denmark drops by 6 positions when we applied our method. Looking at the votes, the weights of supporters for this country are below the average. In particular, we point out that Denmark and Iceland gave 12 to each other. Then both received small weights for their votes, consequently the 12 they exchange was less taken into account in the second ranking (Iceland was 14^{th} and drops by 4 positions).

MovieLens. Our experiment concerns a data set[1] of 100,000 evaluations given by 943 raters on 1682 movies and ranging from 1 to 5. The data have been cleaned so that each rater rates at least 20 movies.
In order to test the robustness of our method, two types of behavior are analyzed in the sequel: first, raters that give random evaluations, and second, spammers that try to improve the reputation of their preferred items.

Random raters. We added to the original data set 237 raters evaluating randomly a random set of items. In that manner, 20% of the raters give random evaluations. Let r^* and $\tilde{r}^*$ be respectively the reputation vector before and after the addition of the random raters. In this configuration, we expect that the random raters will receive smaller weights than the original raters and therefore the vectors of reputations should not be too different.

Fig. 7.11 illustrates the effect of adding random raters for two different methods: first for our method where the total perturbation can be measure by the distance $\|r^* - \tilde{r}^*\|_1 = 182$, and second by taking the average. In this case, the reputations tend to the same averaged value 3. The distance is then given by $\|r^* - \tilde{r}^*\|_1 = 259$ and is naturally greater since random raters receive as much weight as the others.
We can also be interested in the evolution of the weights during the iterations. The distribution of weights is shown in Fig. 7.12 after 1 step, two steps and convergence. We remark that one iteration of the algorithm gives a partial information to trust the raters, it is indeed useful to wait until convergence to have a better separation. This figure also explains how comes the smaller perturbation in Fig. 7.11 when random raters are added comparing to the averaging method; the random raters are about two times less taken into account in our method. Moreover, we see that

[1]The MovieLens data set used in this paper was supplied by the GroupLens Research Project.

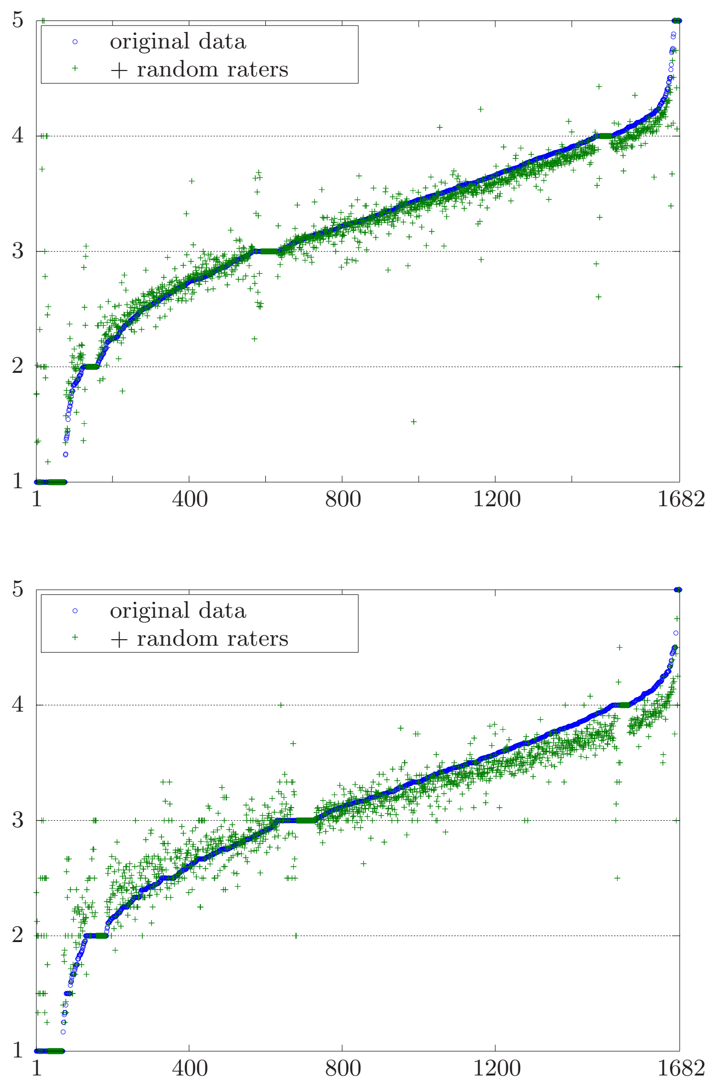


Figure 7.11: X-Axis: the sorted movies according to their reputations before the addition of random raters. Y-Axis: their reputations according to our method (Top) and to the average (Bottom).

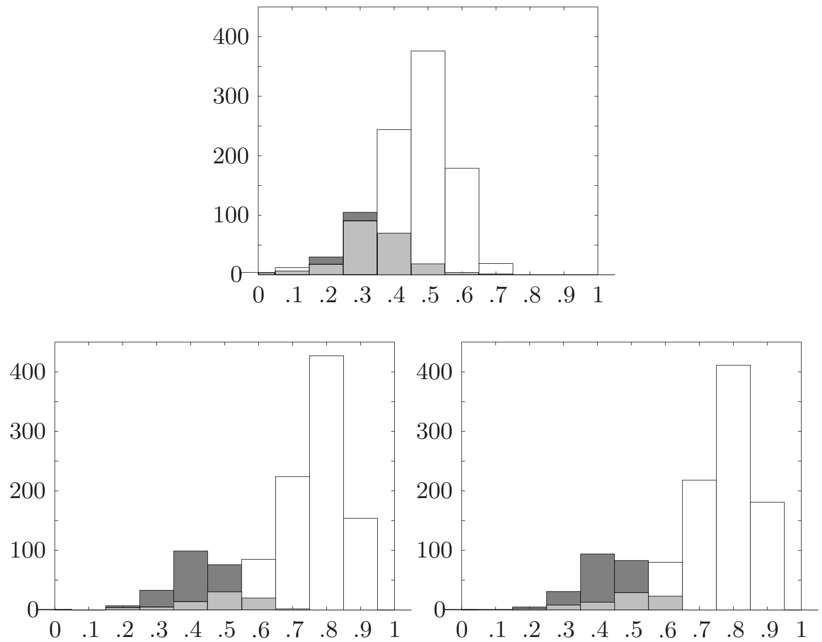


Figure 7.12: X-Axis: the weights of the raters. Y-Axis: the density after one iteration (Top), after two iterations (Left), and after convergence (Right). In black: the random raters. In white: the original raters. In grey: overlap of both raters.

a minority of raters in the original data have weights that are not better than the ones of random raters. Such raters diverge as much as random raters according to our method and therefore they are penalized in a similar fashion.

The parameter k is updated after every iteration step according to Eq. (7.29). After one iteration step, we have $k^1 = 0.05$, then $k^2 = 0.24$ and eventually it converges to 0.23.

Spammers We now added to the original data set 237 spammers giving always 1 except for their preferred movie, which they rated 5. Let r^* and $\tilde{r}^*$ be respectively the reputation vector before and after the addition of these cheaters. Again, we expect that such behavior will be penalized by decreasing the cheater's weights.

Fig. 7.13 illustrates the effect of adding spammers for 2 different methods: first for our method where the total perturbation can be mea-

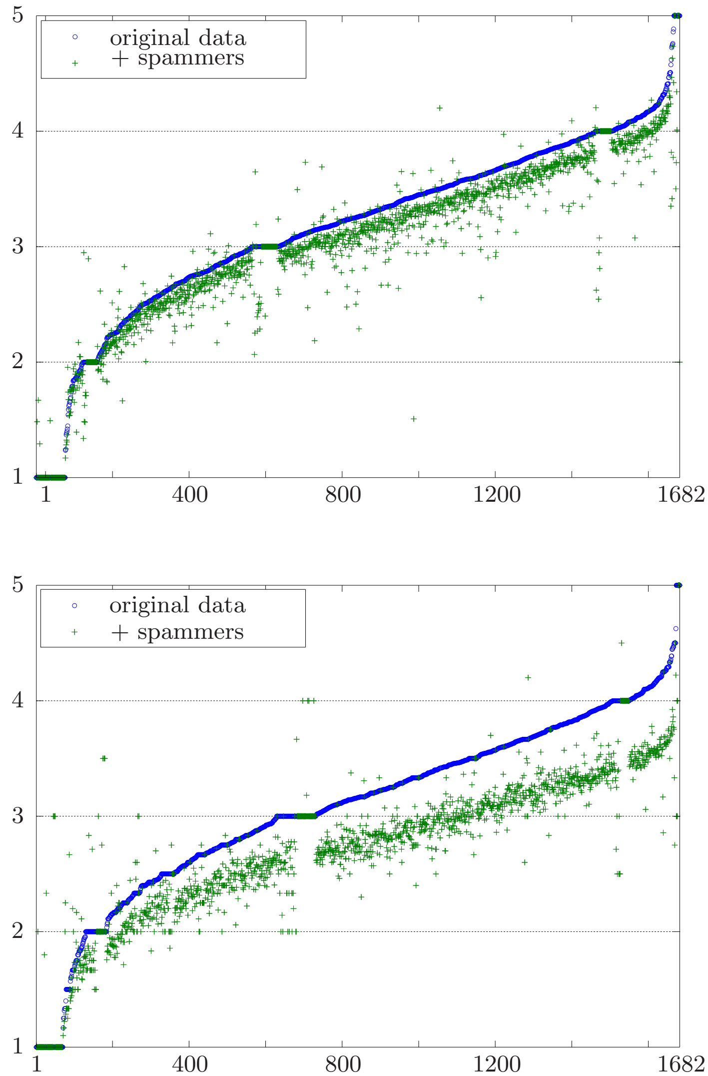


Figure 7.13: X-Axis: the sorted movies according to their reputations before the addition of spammers. Y-Axis: their reputations according to our algorithm (Left) and to the average (Right).

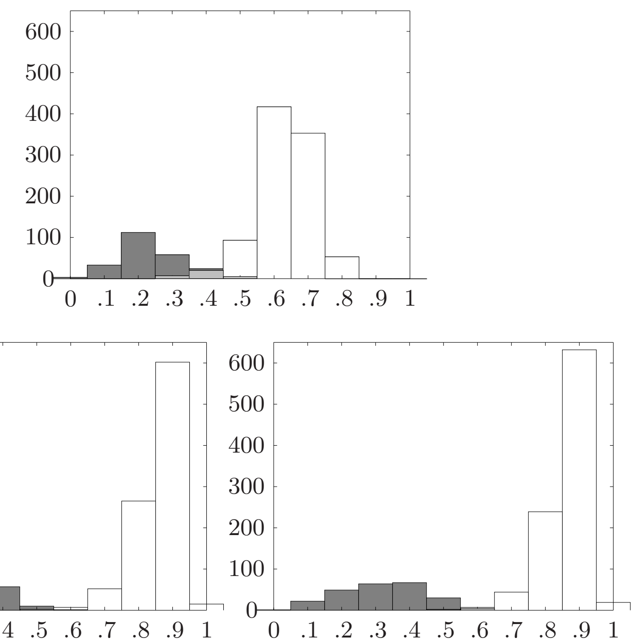


Figure 7.14: X-Axis: the weights of the raters. Y-Axis: the density after one iteration (Top), after two iterations (Left), and after convergence (Right). In black: the spammers. In white: the original raters. In grey: overlap of both raters.

sured by the distance $\|r^* - \tilde{r}^*\|_1 = 267$, and second by taking the average where we see that all reputations tend to be diminished. The distance is then given by $\|r^* - \tilde{r}^*\|_1 = 638$ and this is naturally greater since spammers receive as much weight as the others.

We can also be interested in the evolution of the weights during the iterations. The distribution of weights is shown in Fig. 7.14 after one step, two steps and convergence. As before, we notice that one iteration of the algorithm gives a partial information to trust the raters. Since the spammers are about three times less taken into account, we observed a small difference after their addition in Fig. 7.13. Again a minority of raters in the original data have weights that are not better than the ones of spammers. Such raters diverge as much as spammers according to our method and they are in a similar fashion penalized.

The parameter k is updated after every iteration step according to Eq. (7.29). After one iteration step, we have $k^1 = 0.03$, then $k^2 = 0.13$ and eventually it converges to 0.12.

7.7 Conclusions

Results. The general definition of Iterative Filtering systems provides a new framework to analyze and evaluate voting systems. We emphasized the need for a differentiation of trusts between the raters unlike what is usually done on the Web. The originality of the approach lies in the continuous validation scale for the votes. Next, we assumed that the set of raters is characterized by various possible behaviors including raters who are clumsy or partly dishonest. However, the outliers being in obvious disagreement with the other votes remain detectable by the system as shown in simulations involving alliances, random votes and spammers.

We focus on a natural subclass of IF systems called quadratic IF systems and we show the existence of an energy function that allows us to link a steepest descent to each step of the iteration. It then follows that the system minimizes the belief divergence according to some norm defined from the choice of the discriminant function. We also analyze several discriminant functions; some (the inverse and E-M functions) were already introduced in the literature and others (the exponential and affine functions) are new.

The main effort in the paper concerns the analysis of quadratic IF systems with the choice of the affine discriminant function. This choice is first motivated by a statistical interpretation on the distribution of the votes. The second motivation resides in the direct condition on the parameter k that guaranties the uniqueness of the solution. Moreover this unique solution has the interesting property to maximize the Euclidian norm of the weights of the raters. And last but not least, the analysis of the system, and more precisely its convergence, becomes tractable. We also gave experimental results on real data sets that illustrate the relevance of our approach.

Future research. We see two important application areas of voting systems: first, the general definition of IF systems offers the possibility to analyze various systems depending on the context and the objectives we aim for; second, the experimental tests and the comparisons are crucial to validate the desired properties (including dynamical properties) and to discuss the choice of the IF systems.

Concerning the first part, the quadratic IF systems already give a large choice of different systems via the discriminant function. The candidates we encountered can be extended to other possibilities motivated by some statistical assumptions on the votes, or by the minimization of some energy function (that would also maximize some function on the weights). Moreover, by accepting complex dynamics for evolving votes, we can include multiple solutions in the system. As already mentioned, such equilibria may be interpreted as coherent opinions among groups of people and such methods can be compared with clustering methods. In addition, it remains useful to keep a parameter in the discriminant functions (that is k in our examples) that allows us to adjust the level of separation we like.

Our validation tests can be expanded to other data sets, we think in particular of dynamical data sets that have been hardly investigated for voting systems. Moreover, other behaviors that those we presented are possible. We gave naive profiles to the fictive raters we added in the second simulation, however we know that intermediate behaviors, e.g., half honest and half spammer, are harder to detect. Despite everything, malicious people could use the system to vote correctly several items and then cheat on a few items. Such a strategy can be discouraged when votes are not free (for example, the votes in eBay are made after a transaction), or by tracking these kind of *traders*. In addition to experiments on other data sets with other profiles of raters, the comparison between different IF systems provides another area to pursue. In our case, we gave a toy example with three raters and two ice skaters to compare three discriminant functions. Then, for larger data sets, we limit ourselves to the affine function, the only one that gives a unique solution, and we compared it with the average when some perturbations occur. If we disregard the property of uniqueness for the solution, we can compare our method with others, but we need then to discuss the choice of different initial points that lead to different solutions for these systems. On the other hand, what we want to compare is not trivial since the goal of IF systems is not a simple separation between good and bad raters but rather a continuous validation scale.

Chapter 8

General Conclusions

The aim of this last chapter is to give a general insight of what have been studied through the chapters of this thesis and to see how their topics are connected. We also show that the issues we have treated in this thesis still receive a lot of attention in the scientific literature, and that their solutions meet real needs in viral marketing, in the design of search engines and in the confidence in online markets. We terminate by discussing several fundamental questions.

The thesis. This work has studied the general issue of ranking the nodes of a network. We have seen through the chapters that having a high rank for a node can mean different things like being a leader, being often visited by a surfer or receiving good evaluations from reliable raters.

In Chapter 3, we address the question of finding a ranking vector that reflects the influence of customers on other customers. The originality of the method lies in the fact that we identify the leaders by only looking at their local connections (pairs of friends) in mobile phone networks. Such a measure allows us to be competitive with classical measures that look at the number of contacts or the total number of calls. We also emphasize the importance of a preliminary cleaning of the network where the interpretation of the links is made clear. This refers to the irregular links that do not correspond to any real relationship.

We analyze the local leaders, namely the degree leaders, in random networks in Chapter 4. An interesting phase transition is underlined when the tail of the degree distribution follows a power law. Such distributions are typical in social networks like the mobile phone network studied in Chapter 3. More precisely, being a degree leader undergoes a transition from a *rich gets richer* to a *rich is poor* situation. The results sheds light on the relation between the degree of a node and its probability to be local leader.

In Chapter 5, we return to a well known method of ranking: the PageRank algorithm. That method was used in Chapter 3 to identify leaders and was compared to other methods. We wanted then to study how a set of nodes can choose their outlinks to increase their rank and hence, improve their position in the ranking list. That problem was already studied for a single node, and our work is therefore an extension of previous results for several nodes wishing to maximize the sum of their PageRanks.

The PageRank algorithm does not take into account negative links of the networks. We propose to consider negative linkage by using the PageTrust algorithm presented in Chapter 6. The method is introduced as an extended random walk that will avoid some forbidden nodes depending on negative links. Reconsidering the different manners to maximize one's PageRank in Chapter 3, we show via examples that the PageTrust method is robust. Indeed, there is no easy rule for using negative links in order to increase its own rank.

Finally, Chapter 7 introduces a "double ranking" method where a node has a first rank according to its weighted inlinks (received votes), and a second rank considering its weighted outlinks (given votes). Clearly, when a set of raters evaluates a set of objects, we can expect that not all of them are completely reliable. Therefore the iterative filtering that we propose gives a weight (of credibility) to every rater and then it takes the weighted average of the votes (according to their credibility) to give a reputation score to every object. The advantage of the method (compared to existing methods) lies in the uniqueness of the two ranking vectors (one as raters and one as objects) and in the treatment of dynamical votes.

In progress. As already discussed in Section 1.1 of Chapter 1, the topics studied in this work are still timely. Let us emphasize the main topics in progress.

The measure of social leaders is still successfully used for viral marketing campaigns. Moreover, the identification of leaders in a mobile phone network is now studied with the extra information of antenna locations that give a localization of the calls. On the other hand, we analyze the stability over time of several local measures of leaders such as those presented in Chapters 3 and 4.

The PageRank algorithm remains a hot topic and its robustness against malicious users represents a wide part of the literature. There, the goal is to purchase and delete the so-called spam farms in the Web. They represent meaningless webpages that increase their PageRank by creating artificial structures. Therefore, the optimal structures presented in Chapter 5 gives new patterns for spam detection.

The consideration of negative links in Chapter 6 for a ranking method is relatively new. The difficulty lies in finding a method robust to attacks. Indeed, the negative links may easily be used to disqualify other nodes in the ranking list. Networks with positive and negative links can be found in many areas such as the peer to peer systems and online markets like eBay. For these networks, a growing number of ranking methods is studied and the consideration of negative links become necessary.

Voting systems draw more and more attention these last few years, because of the increasing need of bringing order among the raters surfing in the World Wide Web. The phenomenon of spam is a real issue that can be treated by reputation systems such as those presented in Chapter 7. In particular, the area of peer to peer systems offers a nice application that is often exploited in the scientific literature.

Fundamental questions. We end this thesis by several questions that deserve to be asked for future research on viral marketing, on the PageRank, on negative linkage and on iterative filtering systems.

We mentioned that several marketing campaigns used the definition of social leaders with success. However, the mechanism of viral marketing remains complex and there exists no completely objective way to measure and validate a set of chosen leaders. An interesting discussion on the topic can be found in [102]. There, the authors argue that the phenomenon of cascade in a social network is not necessarily due to a set of leaders (called *influentials* in the article), but may exist because there is *a critical mass of easily influenced individuals*. We believe that this distinction must be kept in mind when interpreting viral marketing results.

Another fundamental question concerns the future of the PageRank algorithm. Its detractors would argue that its use by Google is more and more marginal compared to the more complicated tools developed these last years by the company (e.g., semantic search and personalized search). However, we observe that the PageRank is also used in other contexts as bibliometrics, graduate programs and sport teams. Moreover, that method remains a simple model from which we can start to build or understand other more complex methods. That is what was done in Chapter 6 by considering negative links of a network.

The challenge of finding efficient algorithms for networks with millions or billions of nodes arose about a decade ago. That was at the same time as the arrival of large exploitable databases. Nowadays, more and more databases contain positive and negative links. However, as already mentioned, the negative links are not always taken into account by ranking methods, community detection methods, visualization methods and in general by methods of information retrieval in large networks. We believe that their consideration in these methods is meaningful and provides a wide area of research that will be motivated by the increasing number of large networks containing negative links.

The last question arises in the context of reputation systems where we have rater-rater evaluations or rater-object evaluations. In Chapter 7, we started from the principle that *Raters diverging often from other raters' opinion are less taken into account.* However, such a claim admits that the majority is always right. This underlines the difficulty to detect voting fraud and the risk to follow the majority opinion. Applying iterative filtering methods must therefore be justified according to the context. This leads us to the question of determining the axioms that a reputation systems should satisfy.

Bibliography

[1] G.A. Akerlof. The Market for Lemons: Quality Uncertainty and the Market Mechanism. *Readings in Social Welfare: Theory and Policy*, 2000.

[2] M.S. Aktas, M.A. Nacar, and F. Menczer. Using hyperlink features to personalize web search. *Lecture Notes in Computer Science*, 3932:104, 2006.

[3] R. Albert, H. Jeong, and A.L. Barabási. Internet: diameter of the World-Wide Web. *Nature*, 401(6749):130–131, 1999.

[4] W.H. Au, K.C.C. Chan, and X. Yao. A novel evolutionary data mining algorithm with applications to churn prediction. *Evolutionary Computation, IEEE Transactions on*, 7(6):532–545, 2003.

[5] K. Avrachenkov and N. Litvak. Decomposition of the Google PageRank and optimal linking strategy. Technical report, INRIA, 2004.

[6] K. Avrachenkov and N. Litvak. The effect of new links on Google PageRank. *Stoch. Models*, 22(2):319–331, 2006.

[7] R. Baeza-Yates, C. Castillo, and V. López. PageRank increase under different collusion topologies. In *First International Workshop on Adversarial Information Retrieval on the Web*, 2005. `http://airweb.cse.lehigh.edu/2005/baeza-yates.pdf`.

[8] A.L. Barabási. *Linked: How Everything Is Connected to Everything Else and What It Means for Business, Science, and Everyday Life*. Plume Books, 2003.

[9] A. Berman and R.J. Plemmons. *Nonnegative matrices in the mathematical sciences*, volume 9 of *Classics in Applied Mathematics*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1994.

[10] M. Bianchini, M. Gori, and F. Scarselli. Inside PageRank. *ACM Trans. Inter. Tech.*, 5(1):92–128, 2005.

[11] V.D. Blondel, J.L. Guillaume, J.M. Hendrickx, C. de Kerchove, and R. Lambiotte. Local leaders in random networks. *Physical Review E*, 77(3):36114, 2008.

[12] V.D. Blondel, J.L. Guillaume, R. Lambiotte, and E.L.J.S. Mech. Fast unfolding of communities in large networks. *J. Stat. Mech*, page P10008, 2008.

[13] V.D. Blondel, C. de Kerchove, E. Huens, and P. Van Dooren. Social Leaders in Graphs. *Lecture Notes in Control and Information Sciences*, 341:231, 2006.

[14] M. Boguñá, R. Pastor-Satorras, and A. Vespignani. Absence of Epidemic Threshold in Scale-Free Networks with Degree Correlations. *Physical Review Letters*, 90(2):28701, 2003.

[15] S. Brin and L. Page. The anatomy of a large-scale hypertextual web search engine. *Computer Networks and ISDN Systems*, 30(1–7):107–117, 1998.

[16] PM Brockhoff and IM Skovgaard. Modelling individual differences between assessors in sensory evaluations. *Food quality and preference*, 5(3):215–224, 1994.

[17] D.S. Callaway, MEJ Newman, S.H. Strogatz, and D.J. Watts. Network Robustness and Fragility: Percolation on Random Graphs. *Physical Review Letters*, 85(25):5468–5471, 2000.

[18] M. Catanzaro, M. Boguńá, and R. Pastor-Satorras. Generation of uncorrelated random scale-free networks. *Physical Review E*, 71(2):27103, 2005.

[19] G.M. Del Corso, A. Gullí, and F. Romani. Ranking a stream of news. In *Proceedings of the 14th international conference on World Wide Web*, pages 97–106. ACM New York, NY, USA, 2005.

[20] P.G. Doyle and J.L. Snell. *Random walks and electric networks*, volume 22 of *Carus Mathematical Monographs*. Mathematical Association of America, Washington, DC, 1984.

[21] L. Elden. A note on the eigenvalues of the Google matrix. *Arxiv preprint math/0401177*, 2004.

[22] L.C. Freeman. A set of measures of centrality based on betweenness. *Soc.*, 40(1):35–41, 1977.

[23] L.C. Freeman. Centrality in social networks: Conceptual clarification. *Social Networks*, 1(3):215–239, 1979.

[24] S. Galam. Application of statistical physics to politics. *Physica A*, pages 132–139, December 1999.

[25] J. Galambos. The Asymptotic Theory of Extreme Order Statistics. *New York*, 1978.

[26] V.A. Ginsburgh and A. Noury. Cultural Voting: The Eurovision Song Contest. 2004.

[27] M.C. Gonzales, A.H. César, and A.L. Barabási. Understanding individual human mobility patterns. *Nature*, 453:779–782, 2008.

[28] A. Granas and J. Dugundji. *Fixed point theory*. Springer, 2003.

[29] R. Guha, R. Kumar, P. Raghavan, and A. Tomkins. Propagation of trust and distrust. In *Proceedings of the 13th international conference on World Wide Web*, pages 403–412. ACM Press New York, NY, USA, 2004.

[30] Z. Gyöngyi and H. Garcia-Molina. Link spam alliances. In *VLDB '05: Proceedings of the 31st international conference on Very large data bases*, pages 517–528. VLDB Endowment, 2005.

[31] Z. Gyöngyi, H. Garcia-Molina, and J. Pedersen. Combating web spam with TrustRank. In *Proceedings of the Thirtieth international conference on Very large data bases-Volume 30*, pages 576–587. VLDB Endowment, 2004.

[32] D. Harel and Y. Koren. A fast multi-scale method for drawing large graphs. *Graph Algo. Appl.*, 6(3):179–202, 2002.

[33] T.H. Haveliwala and S.D. Kamvar. The second eigenvalue of the Google matrix. *A Stanford University Technical Report http://dbpubs. stanford. edu*, 8090, 2003.

[34] C.A. Hidalgo and C. Rodriguez-Sickert. The dynamics of a mobile phone network. *Physica A: Statistical Mechanics and its Applications*, 387(12):3017–3024, 2008.

[35] S. Hill, F. Provost, and C. Volinsky. Network-based marketing: Identifying likely adopters via consumer networks. *Statistical Science*, 21(2):256, 2006.

[36] R.A. Horn and C.R. Johnson. *Matrix analysis*. Cambridge university press, 1985.

[37] I.C.F. Ipsen and T.M. Selee. PageRank Computation, with Special Attention to Dangling Nodes. *SIAM Journal on Matrix Analysis and Applications*, 29(4):1281–1296, 2007.

[38] I.C.F. Ipsen and R.S. Wills. Mathematical properties and analysis of Google's PageRank. *Bol. Soc. Esp. Mat. Apl.*, 34:191–196, 2006.

[39] I.C.F. Ipsen and R.S. Wills. Ordinal Ranking for Google's PageRank. *SIAM Journal on Matrix Analysis and Applications*, 30(4):1677–1696, 2009.

[40] H. Jeong, Z. Neda, and A.L. Barabasi. Measuring preferential attachment in evolving networks. *Europhysics Letters*, 61(4):567–572, 2003.

[41] T. Kamada and S. Kawai. An algorithm for drawing general undirected graphs. *Information Processing Letters*, 31(1):7–15, 1989.

[42] S.D. Kamvar, M.T. Schlosser, and H. Garcia-Molina. The Eigentrust algorithm for reputation management in P2P networks. In *Proceedings of the 12th international conference on World Wide Web*, pages 640–651. ACM Press New York, NY, USA, 2003.

[43] J.G. Kemeny and J.L. Snell. *Finite Markov chains*. The University Series in Undergraduate Mathematics. D. Van Nostrand Co., Inc., Princeton, N.J.-Toronto-London-New York, 1960.

[44] D. Kempe, J. Kleinberg, and É. Tardos. Maximizing the spread of influence through a social network. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 137–146. ACM New York, NY, USA, 2003.

[45] C. de Kerchove, G. Krings, R. Lambiotte, P. Van Dooren, and V.D. Blondel. Role of second trials in cascades of information over networks. *Physical Review E*, 79(1):6114, 2009.

[46] C. de Kerchove, L. Ninove, and P. Dooren. Maximizing PageRank via outlinks. *Linear Algebra and Its Applications*, 429(5-6):1254–1276, 2008.

[47] C. de Kerchove and P. Van Dooren. Reputation systems and optimization. *Siam News*, 41(2), 2008.

[48] C. de Kerchove and P. Van Dooren. The PageTrust algorithm: how to rank web pages when negative links are allowed. In *Proceedings of the Siam International Conference in Data Mining*. SIAM, 2008.

[49] C. de Kerchove and P. Van Dooren. Iterative filtering in reputation systems. *submitted to SIAMAX*, 2009.

[50] S. Kirkland. Conditioning of the entries in the stationary vector of a Google-type matrix. *Linear Algebra Appl.*, 418(2-3):665–681, 2006.

[51] J.O. Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM*, 46(5):604–632, 1999.

[52] E. Kotsovinos, P. Zerfos, N.M. Piratla, N. Cameron, and S. Agarwal. Jiminy: A Scalable Incentive-Based Architecture for Improving Rating Quality. *Lecture Notes in Computer Science*, 3986:221, 2006.

[53] P.L. Krapivsky and S. Redner. Statistics of Changes in Lead Node in Connectivity-Driven Networks. *Physical Review Letters*, 89(25):258703, 2002.

[54] P.L. Krapivsky and S. Redner. Network growth by copying. *Physical Review E*, 71(3):36118, 2005.

[55] R. Lambiotte. How does degree heterogeneity affect an order-disorder transition? *EPL*, 78(6):68002, 2007.

[56] R. Lambiotte and M. Ausloos. Growing network with j-redirection. *EPL*, 77(5):58002, 2007.

[57] R. Lambiotte, V.D. Blondel, C. de Kerchove, E. Huens, C. Prieur, Z. Smoreda, and P. Van Dooren. Geographical dispersal of mobile communication networks. *Physica A: Statistical Mechanics and its Applications*, 2008.

[58] Amy N. Langville and Carl D. Meyer. Deeper inside PageRank. *Internet Math.*, 1(3):335–380, 2004.

[59] A.N. Langville and C.D. Meyer. A Survey of Eigenvector Methods for Web Information Retrieval. *SIAM Review*, 47(1):135–161, 2005.

[60] A.N. Langville and C.D. Meyer. *Google's PageRank and beyond: the science of search engine rankings*. Princeton University Press, Princeton, NJ, 2006.

[61] M. Latapy. Main-memory triangle computations for very large (sparse (power-law)) graphs. *Theoretical Computer Science*, 407(1-3):458–473, 2008.

[62] P. Laureti, L. Moret, Y. Zhang, and YK Yu. Information filtering via Iterative Refinement. *Europhysics Letters*, 75(6):1006–1012, 2006.

[63] R. Lempel and S. Moran. The stochastic approach for link-structure analysis (SALSA) and the TKC effect. *Computer Networks*, 33(1-6):387–401, 2000.

[64] R. Lempel and S. Moran. Rank-stability and rank-similarity of link-based web ranking algorithms in authority-connected graphs. *Inf. Retr.*, 8(2):245–264, 2005.

[65] J. Leskovec, L.A. Adamic, and B.A. Huberman. The dynamics of viral marketing. *ACM Press New York, NY, USA*, 2007.

[66] D.D. Lewis. Naive (Bayes) at forty: The independence assumption in information retrieval. *Lecture Notes in Computer Science*, 1398:4–18, 1998.

[67] T. Luczak and P. Erdos. Changes of leadership in a random graph process. *Random Struct. Algorithms*, 5:243–252, 1994.

[68] S.A. Macskassy and F. Provost. Classification in networked data: A toolkit and a univariate case study. *The Journal of Machine Learning Research*, 8:935–983, 2007.

[69] N. Madras and G. Slade. *The self-avoiding walk*. Birkhauser, 1996.

[70] P. Massa and P. Avesani. Controversial Users Demand Local Trust Metrics: An Experimental Study on Epinions. com Community. In *Proceedings of the National Conference on Artificial Intelligence*, volume 20, page 121, 2005.

[71] P. Massa and C. Hayes. Page-reRank: Using Trusted Links to Re-Rank Authority. In *Web Intelligence, 2005. Proceedings. The 2005 IEEE/WIC/ACM International Conference on*, pages 614–617, 2005.

[72] R.M. May and A.L. Lloyd. Infection dynamics on scale-free networks. *Physical Review E*, 64(6):66112, 2001.

[73] A.O. Mendelzon and D. Rafiei. An autonomous page ranking method for metasearch engines. In *The Eleventh International WWW Conference, May*, 2002.

[74] C.D. Meyer. *Matrix Analysis and Applied Linear Algebra*. Society for Industrial Mathematics, 2000.

[75] A.A. Moreira, J.S. Andrade Jr, and L.A. Nunes Amaral. Extremum Statistics in Scale-Free Network Models. *Physical Review Letters*, 89(26):268703, 2002.

[76] L. Mui, M. Mohtashemi, and A. Halberstadt. A computational model of trust and reputation. In *Proceedings of the 35th Hawaii International Conference on System Sciences*, pages 188–196, 2002.

[77] M.E.J. Newman. Assortative Mixing in Networks. *Physical Review Letters*, 89(20):208701, 2002.

[78] M.E.J. Newman. The structure and function of complex networks. *SIAM Rev.*, 45(2):167–256, 2003.

[79] M.E.J. Newman. Who Is the Best Connected Scientist? A Study of Scientific Coauthorship Networks. *Lecture Notes in Physics*, 650:337–370, 2004.

[80] MEJ Newman. Modularity and community structure in networks. *Proceedings of the National Academy of Sciences*, 103(23):8577–8582, 2006.

[81] A.Y. Ng, A.X. Zheng, and M.I. Jordan. Stable algorithms for link analysis. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 258–266. ACM New York, NY, USA, 2001.

[82] L. Ninove. *Dominant vectors of nonnegative matrices. Application to information extraction in large graphs*. PhD thesis, Université catholique de Louvain, 2008.

[83] J. O'Donovan and B. Smyth. Trust in recommender systems. In *Proceedings of the 10th international conference on Intelligent user interfaces*, pages 167–174. ACM New York, NY, USA, 2005.

[84] J.P. Onnela, J. Saramaki, J. Hyvonen, G. Szabo, D. Lazer, K. Kaski, J. Kertesz, and A.L. Barabasi. Structure and tie strengths in mobile communication networks. *Proceedings of the National Academy of Sciences*, 104(18):7332, 2007.

[85] L. Page, S. Brin, R. Motwani, and T. Winograd. The PageRank citation ranking: Bringing order to the Web. Technical report, Computer Science Department, Stanford University, 1998. `http://dbpubs.stanford.edu:8090/pub/1999-66`.

[86] R. Pastor-Satorras and A. Vespignani. Epidemic Spreading in Scale-Free Networks. *Physical Review Letters*, 86(14):3200–3203, 2001.

[87] J. Patel, W.T.L. Teacy, N.R. Jennings, and M. Luck. A probabilistic trust model for handling inaccurate reputation sources. *Lecture Notes in Computer Science*, 3477:193–209, 2005.

[88] P. Resnick, R. Zeckhauser, J. Swanson, and K. Lockwood. The value of reputation on eBay: A controlled experiment. *Experimental Economics*, 9(2):79–101, 2006.

[89] M. Richardson, R. Agrawal, and P. Domingos. Trust Management for the Semantic Web. *Lecture Notes in Computer Science*, pages 351–368, 2003.

[90] M. Richardson and P. Domingos. The Intelligent Surfer: Probabilistic Combination of Link and Content Information in PageRank. *Advances in Neural Iformation Processing Systems*, 2:1441–1448, 2002.

[91] H. Schneider. The influence of the marked reduced graph of a nonnegative matrix on the Jordan form and on related properties: a survey. *Linear Algebra and its Applications*, 84:161–189, 1986.

[92] E. Seneta. *Nonnegative matrices and Markov chains.* Springer Series in Statistics. Springer-Verlag, New York, second edition, 1981.

[93] S. Serra-Capizzano. Jordan canonical form of the Google matrix: A potential contribution to the PageRank computation. *SIAM Journal on Matrix Analysis and Applications*, 27:305, 2005.

[94] Z. Smoreda and C. Licoppe. Effets du cycle de vie et des réseaux de sociabilité sur la téléphonie. *Rapport, Issy-les-Moulineaux, CNET*, 1998.

[95] V. Sood and S. Redner. Voter Model on Heterogeneous Graphs. *Physical Review Letters*, 94(17):178701, 2005.

[96] H.E. Stanley and V.K. Wong. Introduction to Phase Transitions and Critical Phenomena. *American Journal of Physics*, 40:927, 1972.

[97] M. Sydow. Can one out-link change your PageRank? In *Advances in Web Intelligence (AWIC 2005)*, volume 3528 of *Lecture Notes*

in Computer Science, pages 408–414. Springer Berlin Heidelberg, 2005.

[98] G. Szabo and A. . Barabasi. Network effects in service usage. *ArXiv Physics e-prints*, November 2006.

[99] G. Theodorakopoulos and J.S. Baras. On trust models and trust evaluation metrics for ad hoc networks. *IEEE Journal on Selected Areas in Communications*, 24(2):318–328, 2006.

[100] V. A. Traag and Jeroen Bruggeman. Community detection in networks with positive and negative links. 2008.

[101] P.K. Varshney and CS Burrus. *Distributed detection and data fusion*. Springer Verlag, 1997.

[102] D.J. Watts and P.S. Dodds. Influentials, networks, and public opinion formation. *Journal of Consumer Research*, 34(4):441–458, 2007.

[103] JL Willems. Stability theory of dynamical systems. *Studies in Dynamical Systems*, 1970.

[104] C.F.J. Wu. On the convergence properties of the EM algorithm. *The Annals of Statistics*, 11(1):95–103, 1983.

[105] Yi-Kuo Yu, Yi-Cheng Zhang, Paolo Laureti, and Lionel Moret. Decoding information from noisy, redundant, and intentionally-distorted sources, 2006.

[106] S. Zhang, Y. Ouyang, J. Ford, and F. Makedon. Analysis of a low-dimensional linear model under recommendation attacks. In *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 517–524. ACM New York, NY, USA, 2006.