Université Pierre et Marie Curie

— **UPMC** — Paris 6 —

HABILITATION TO SUPERVISE RESEARCH

Specialty : **Computer Science**
Domain : **Networking and Internet Architecture (cs.NI)**

On Peer-to-peer: Content Distribution, Acyclic Preference Networks

by Fabien Mathieu

Defended on February 11, 2009 before a committee of:

M. Pascal Felber, Professor, University of Neuchâtel............Reviewer
M. Pierre Fraigniaud, Research Director, CNRS............Reviewer
M. Jérôme Galtier, Orange Labs............Examiner
M. Laurent Massoulié, Thomson Technology Paris Laboratory............Examiner
M. Philippe Robert, Research Director, INRIA............Reviewer
M. Sébastien Tixeuil, Professor, Pierre et Marie Curie University............Chair
M. Laurent Viennot, Research Scientist, INRIA............Examiner

Disclaimer 2026

I recently converted my doctoral thesis, defended in 2004, from LaTeX to Typst format. The original motivation was very simple: I could no longer compile the original LaTeX sources.

The source files for my habilitation thesis (HDR), written in late 2008 and defended in February 2009, still compile. However, I also wanted to convert them, in order to have a more modern version available in English and in HTML.

I chose to preserve the original document's content as much as possible, while adapting the formatting to Typst's capabilities. Obvious typos have been corrected, but broken URLs have been left as they were. This is not a new version of the thesis, but a faithful reproduction of the original document in a new typesetting system. The goal is to make the thesis content more accessible and easier to maintain, while preserving the original ideas and arguments as closely as possible.

All peers are equal,
but some peers are more equal than others.

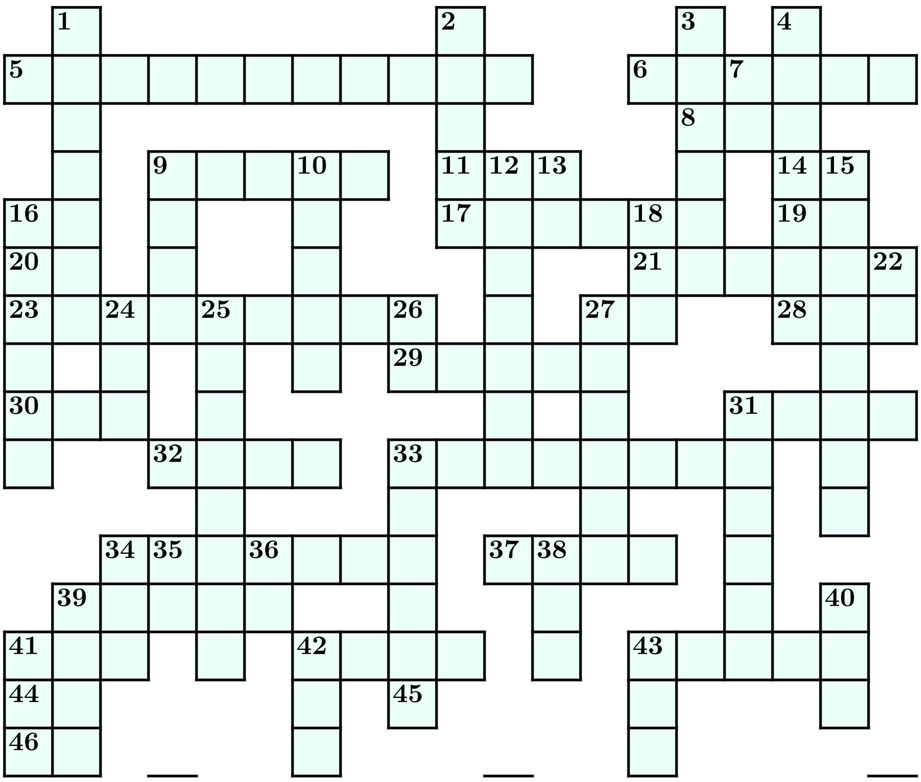
GEORGE ORWELL, *the server farm*

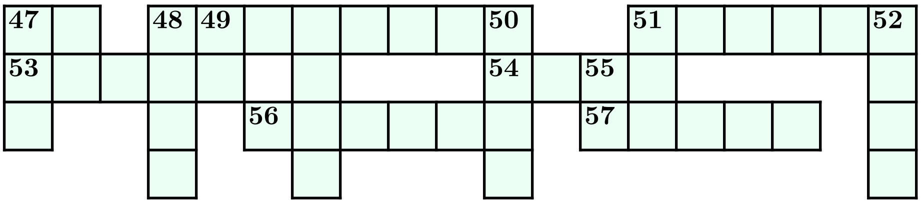
Acknowledgements

The acknowledgements are certainly the most delicate part of a thesis to write. Since my father was a great crossword enthusiast, I decided to pay him this tribute, transferring in the process the difficulty from the author to the reader. A few useful pointers to tackle these acknowledgements in the best possible conditions:

- Many definitions refer to personal anecdotes. It is therefore normal not to be able to solve the entire grid (in this respect, these acknowledgements are after all like any other).
- I do hope, however, that everyone will find their squares. If not, $V31$.
- For living organisms, I only give the first name or the species. In case of a duplicate, a single entry on the grid, but possibly two definitions.
- Some words in the grid are not acknowledgements, but are just there to decrease the difficulty (or increase it according to some). They are often (but not always) the shortest ones.

Note: The crossword puzzle below is in French and relies on wordplay, personal anecdotes, and cultural references that are inherently untranslatable. It is preserved in its original French form. Francophone readers will appreciate the clues; others are warmly invited to seek help from a French-speaking friend.





Across

5. C'est vilain de le faire, mais je suis content qu'ils aient accepté.
6. Où l'on chasse les têtes.
8. Grosse souris sans fil.
9. Annexe parisienne de Montpellier.
11. Total.
14. Pas Dell.
16. Où l'on inverse la notation.
17. Atos (l'autre).
19. Système d'autoroutes aux USA.
20. Prénom balafré.
21. Zen.
23. Expert démoniaque.
27. Messe du soir.
28. Rue des grosses têtes.
29. Calamité.
30. Melmacien.
31. S'est fait avoir.
32. Honorable.
33. Blonde insulaire.
34. Fumeur de Drums.
37. Association criminelle se réunissant le MARDI.
39. Fruit indigeste pour les pandas.
41. C'est le meilleur qui s'en va en premier.
42. Là quand on a besoin d'eux.
43. Apprentice.
44. Fin de soirée.
45. Avec ça, on gagne bien sa vie.
46. Au cœur de l'horreur.
47. Rêve de chercheur.
48. Amateurs de plaisir solitaire.
51. Il faut le voir pour le croire.
53. Jase.
54. Faire fait.

Down

1. C'est compliqué.
2. Motif de plainte.
3. Ma hyène.
4. Un homme à marier ?
7. Sous-sol.
9. Made in Italia.
10. What else ?
12. Push and Beyond – Master.
13. Quinze fois rien.
15. Grand prénom du rap – Amateur de péplums.
16. Le petit se fait rouler (mais pas lui !).
18. 6 heures du mat' les nuits de deadline.
22. Mesure un moment.
24. Tombeau présidentiel.
25. Muridé informaticien (qu'il repose en paix sur son petit nuage de gruyère).
26. Aux normes.
27. D'artagnan.
31. À ceux qui recherchent vainement leur nom ici (cette marque d'affection me va droit au cœur).
33. Le monde de Neo.
34. Ce n'est pas la taille qui compte.
35. Cocotte.
36. Un bleu qui n'a pas lu.
38. Baba cool.
39. Routeur glouton.
40. Expert de l'égalité.
41. Master of the WebWorld.
42. Aramis.
43. White Russian.
48. Vue imprenable sur la Tour Eiffel.
49. Additif.
50. Liquide mis en banque.

56. Une Pomme tombée sur ma tête (merci d'être velu).
57. Canadienne sans fil.
52. Sans hache, j'écris ton nom.
55. À fond de train.

Foreword

The scene takes place in a drinking establishment, on a long afternoon of a waning summer month. Two friends are lounging on the terrace. The beer is cold, and the conversation light. The subject of debate: the existence of a soulmate and the quantitative measurement of love. For one of the two protagonists, the space of feelings is an extremely complex geometry that admits no absolute extremum. Adultery is then nothing more than the amorous application of the optimization method known as *simulated annealing*. One thing leading to another, the topic drifts to a *science et vie* article that deals with a mathematical model that gives, among other things, a very tantalizing vision of male-female relationships.

This is how I became acquainted, in August 2005, with the theory of stable marriages. And how I unwittingly found my main research project for the next three years, which would eventually motivate this habilitation thesis.

This leads me to note the analogy between research and skiing: on a known and well-traveled domain, there is little risk of getting lost. The terrain is generally (more or less) well-marked, and it is not very difficult to know which runs are interesting to take.

Conversely, an unknown domain provides the joy of plunging into the powder, where the researcher's hand has never set foot. But one must admit that one generally does not quite know where one is going.

I have thus seen theorems of extraordinary power let themselves be proven in a few minutes, while others have resisted (and still resist), sleepless night after sleepless night, every assault endured. Who has never dreamed of these terrain graphs, of these distant peers populated by legends or of a sudden wealth that would be conquered at the turn of a path of a Kieschnick chain? Who has never wished to see non-local equations guide their steps, at the heart of a stable configuration, toward the riches and history of the mysterious acyclic distributions?

But I am already straying from the foreword to get into the thick of the subject. I hope the reader will take as much pleasure reading this thesis as I took writing it. Be warned, however, that while the introduction was written to be accessible to the widest possible audience, one may encounter throughout the chapters some formulas and proof ideas. In particular during Chapter 4.

Table of Contents

Chapter 1
Introduction

This thesis presents a major part of the research I have conducted since my doctoral thesis, defended in December 2004. It is centered around peer-to-peer (P2P), with special attention dedicated to the topic of preference networks. My work outside the P2P field, mainly research around Google's PageRank, is not developed here (references are available in the Curriculum Vitae attached to this thesis). The two main reasons for this editorial choice are the desire to present a body of work articulated around a common theme (P2P), as well as the interesting constraint posed by writing a thesis of limited size.

In this introductory chapter, I propose to share with the reader my conception of peer-to-peer, not as common sense understands it, but as a research domain. The following chapter will then present my positioning as a researcher within this domain and the structure of the rest of this thesis.

1.1 A subjective history of peer-to-peer

There exists a plethora of works trying to answer the question of what peer-to-peer is and how to approach it. The size, technicality, and medium are extremely variable: books [70, 72], online studies and wikibooks [69, 77], pedagogical introduction [76]...

Given all of this, it is difficult to offer a vision that has not already been proposed elsewhere. This is why I have chosen to present peer-to-peer starting from its emergence as a social phenomenon, viewed through the lens of my personal experience. I do not claim that my approach is recommended for someone who would like to learn about peer-to-peer: it is better in that case to start with one of the references above. Nevertheless, I believe that approaching things "through the wrong end of the telescope," as I am about to do, allows for a better understanding of the path I have followed in my research, and thus brings out what unifies it.

My own analysis of the history of peer-to-peer is therefore the following: I tend to think that peer-to-peer was born at the moment when it was the only technical solution capable of meeting a social need. More precisely, let us go back about ten years or so. The democratization of Internet access, coupled with significant advances in signal processing (known to the general public through the acronyms MP3 and DiVX) was beginning to make it possible to retrieve audio and video "content" within reasonable timeframes.

Problem: where to retrieve the content from? At the time (late 90s), the digital economy was still in its infancy. Traditional media were still hesitant to take the digital leap, and there existed bit[1] of a portal allowing access to content.

Initially, relatively old methods were employed within restricted circles of *underground* users: use of *FTPz* (hijacking an anonymous FTP server in order to use the storage and bandwidth resources of said server), attachments in discussion forums (the *alt.binaries* of Usenet), …

Then, with the Internet bubble came online storage sites. This solution was very quickly preferred over *FTPz*: no more need to scan IP addresses, and lower risk of data loss. The most famous example of this era must be the site http://MySpace.com, which is not at all the current social networking site, but offered 300 MB of storage per account, with no limit on the number of accounts or bandwidth[2].

Alas, one day the bubble burst. Sites like http://MySpace.com closed, or were completely throttled in the best case. The absence of a *business model* around an unlimited and free storage offering, and the astronomical costs generated by abusive bandwidth usage were certainly not unrelated to this. But a portion of Internet users had developed a taste for easy access to content, and did not want to go back. The need was there. It was then that, across various forums, an idea began to emerge: *What if we pooled our storage and bandwidth resources? With enough participants, we would have more resources than any server, wouldn't we?*.

A first attempt, with the software *Napster*, had already been cut short due to legal problems. Nevertheless, driven by the need for content access, software like *KaZaA* or *eDonkey* [2, 3] began to emerge, while somewhere on the web, an unknown computer scientist named Bram Cohen was working on a file-sharing protocol and using files restricted to those over 18 to test what would become BitTorrent [23][3]. It was 2001 and peer-to-peer (P2P) was born from the ashes of FTPz, MySpace, and Napster.

I must admit that when I first heard about this idea of using the users, my reaction was to think *that will never work*. I did not believe that users were ready to offer their bandwidth, with all the constraints that implied. Moreover, I had hastily estimated the performance of such a system using bandwidth conservation, and I had concluded that performance would be extremely poor. Since then, I have had time to realize my error and make amends, as I will discuss in more detail in Chapter 3.

[1] A neo-grammaticalization inspired by the history of *negation in French*.

[2] http://web.archive.org/web/20001019040949/www.myspace.com/Index.asp

[3] I had stumbled by chance upon Bram Cohen's test page in 2004, but I am unable to find it again today, and I can therefore only rely on my memory alone to report this fact. I do not even remember the exact title of the work offered for download, other than it must have been opus 4 or 6 of a themed series.

Today, the success of P2P file-sharing applications, which use only the resources provided by the users, is no longer in question. The time has come for the development of new functionalities, such as *streaming*. Live broadcast applications (or rather slightly delayed broadcast) like CoolStreaming [78], PPLive [41], SopCast [7], TVants [8], or UUSee [9] are becoming increasingly popular, while *video-on-demand* broadcasting is beginning to appear [1, 4, 5].

1.2 Defining peer-to-peer

Now that the context is set, we can begin to reflect on the question of defining peer-to-peer. To illustrate the problem, allow me to recount the following anecdote: there has been much talk recently about the so-called *DADVSI* law relating to *copyright and neighboring rights in the information society*. One of the goals of this law was to counter illegal uses of P2P, or even potentially illegal P2P software. But to legislate, one must define, and this is what article 21 attempts to do, making punishable by three years of imprisonment and 300,000 euros in fines the act of *publishing, making available to the public, or communicating to the public, knowingly and in any form whatsoever, software manifestly intended for the unauthorized making available to the public of protected works or objects*.

This article generated much controversy, notably because as written, it included most Web server software, including the very well-known Apache software. It was thus necessary to exclude these software packages in an ad-hoc manner by adding a special clause. Nevertheless, I find the definition of P2P that emerges from this legal text interesting: P2P software is intended to make content available, the meaning of content being a priori as broad as possible.

This first definition, which I will call a *motivation* definition, is of course insufficient to define P2P, as the DADVSI law clearly demonstrated. One must therefore look further. This is generally where the definition by *opposition* comes in: contrary to the classic client/server paradigm, where each component of the system is either a server (content distributor) or a client (content consumer), the P2P paradigm considers a system where each participant can be both client and server simultaneously.

Much more targeted than the motivation definition, the definition by opposition to the client/server model nevertheless contains a flaw when one seeks to specify what it means to *be able to be both client and server*. Consider for example the *ssh* protocol, a network protocol allowing a client to connect to a server. If I open an ssh session from my computer to a friend's computer, it is clear that I am not doing peer-to-peer. If that friend opens in return an ssh session on my own computer, each will potentially have access to the other's resources (each computer being both client and server), yet this still does not resemble peer-to-peer in the common sense. But if an entire social network starts using ssh sessions to exchange content[4], it ends up forming a peer-to-peer network, or more precisely a *friend-to-friend* network in this specific case. The question is: where is the boundary?

Conversely, it is very easy to imagine client/server uses for P2P software. Thus, every week, I use the BitTorrent client to retrieve MP3s from a friend[5]. He has the files, I retrieve them, BitTorrent serving only to facilitate the transfer: the information needed to retrieve the files is concentrated in a small `.torrent` file that is very easy to produce and exchange by email. Here again, the boundary between the client/server paradigm and P2P is extremely blurry.

It therefore appears that even though it is a good approximation, the definition by opposition is imperfect. Just as one cannot define the essence of peer-to-peer as being tied to a technology, there is no simple architectural categorization (paradigm) into which peer-to-peer fits.

All of this explains why over time I have forged my own definition of peer-to-peer, which tries to best capture the fact that in my experience, P2P is first and foremost a decentralized way of approaching a distribution problem in a given hardware scenario. In other words, if one looks at the evolution of P2P from its beginnings to the present day, one sees a basic principle remaining the same beyond technological evolutions: if one wants to offer any service to a large number of users while having few or no servers at one's disposal, the only solution is to take the resources from the only place where they exist, namely from the users (the peers) themselves.

This observation leads to a third and final definition of P2P, which I would describe as a definition by *complexity*: a system, a usage, a scenario is of the *peer-to-peer* type as soon as the way to answer the question
Who retrieves what from whom?
is non-trivial.

This definition may seem strange and counter-intuitive, but its advantages are certain: it frees itself from technological questions and transforms a semantic question (*what is peer-to-peer?*) into a more algorithmic problem. Thus, to know whether an assembly of ssh connections forms a P2P system, the definition I propose answers *yes, provided that the system is more complex than a succession of independent bilateral distributions*[6]. In other words, one can speak of peer-to-peer as soon as the search for a given file, or its transfer, uses more than one client/server connection.

One also obtains, and this is what interests me as a researcher, a way of defining what peer-to-peer research is: trying to understand the question *who retrieves what from whom* when it becomes complicated, decomposing it into sub-questions more

[4]This is the operating principle of existing software, such as for example *Waste*

[5]I should specify that these are working recordings from rehearsals of the jazz band I am part of, therefore just as legal as they are unlistenable to the public.

[6]As an exercise, I leave it to the reader to have fun answering the question *Is Skype a peer-to-peer application?* using the three definitions I have just given.

specific to a given problem (search and distribution, availability and volatility, fairness and heterogeneity,...) and trying to answer them in the best possible way.

1.3 Research themes

Now that the subject of peer-to-peer research is open, I propose to survey the different themes and approaches on which a P2P researcher may work. I propose to employ a dichotomous classification based on four axes: localization/distribution, structured/ unstructured, push/pull, theoretical/empirical. This classification is far from unique and it is not perfect, but I find it relatively simple and practical.

1.3.1 Localization and distribution

The first way to decompose the question *Who retrieves what from whom?* is to separate this question into two sub-problems: localization and distribution. The localization phase consists of answering the question *Who has what?*: this is the localization phase. One solution is for example that a central server monitors all content present in a system, thus allowing any search to be answered. This is for example the operating principle of BitTorrent *trackers* [23] or *eDonkey* servers [3]. Of course, these solutions are centralized, and therefore not peer-to-peer (the question *who has what* is then trivial). A first step towards complexity is the use of several servers instead of just one (super-nodes). This reduces the concentration of information, which can be dangerous from a security standpoint, but one must begin to think about communication between super-nodes: we have a semi-centralized, semi-complex, in short semi-peer-to-peer solution. Finally, one can let the peers sort things out entirely among themselves to know who has what. Among these peer-to-peer approaches to localization, there is for example *flooding*, which consists of emitting a content request and passing it from neighbor to neighbor, but the most widely used approach is undoubtedly that of *distributed hash tables* [34, 61, 64, 68, 73, 79], which associate each content with a peer (or even several) responsible for knowing who possesses that content.

Once the content is localized, the problem of its effective distribution arises: how to retrieve the content? This is a problem that varies enormously depending on what one means by distribution: downloading a small or large file, viewing a channel with slight delay, a video-on-demand... The subtleties related to the different types of distribution will be explored in depth in Chapter 3. For now, one should simply remember that since the exact problem to be solved varies, the palette of solutions is broad: broadcast trees, use of queues, reciprocal exchanges...

Most of the time, these two problems, localization and distribution, are independent. For example, while localization in a BitTorrent system was historically provided by a centralized tracker, recent versions also offer the possibility of using a

DHT. Similarly, the choice of distribution protocol can adapt based on the result of localization: this is the multi-protocol approach, employed by MLDonkey [30].

Of course, there are exceptions, and the distinction between localization and distribution is not always airtight. Thus, for security and confidentiality reasons, the Freenet software [22] is designed so that distribution follows the same network path as the one created during localization, making the two problems inseparable.

1.3.2 Structured, unstructured

Another way to approach the *who-what-from-whom* question is to think about how the problem will be solved in practice. From this perspective, two major currents exist: structured and self-structured approaches.

The principle of structured approaches is to ensure that peers conform to a pre-determined structure. This structure will guide the peers in their behavior and their manner of exchanging messages or content. Most distributed hash tables use a structured algorithm [34, 61, 64, 68, 73, 79]. Structured solutions have also been proposed to solve the problem of slightly delayed broadcast [19, 35].

These structures are generally graphs (trees, trees with disjoint interior nodes [19], de Bruijn graphs [34, 35], regular graphs, …), sometimes built from a virtual topology (ring [61, 73], d-torus [64], hypercube [61, 68, 73, 79], …).

The strength of structured approaches is that the system's behavior is determined by a structure known in advance, and that this structure can be designed to guarantee optimal, or near-optimal, performance. Their weakness comes from the difficulties of maintaining a structure in the face of peer volatility. Moreover, many structures assume that peers have similar capacities, and with rare exceptions [75], they adapt poorly to a population that is often heterogeneous in practice.

The other solution is to let (more or less) the peers sort things out, that is, let them make decisions based on their own perception of the system. There is then no longer an explicit structure, but a fuzzy and implicit pseudo-structure resulting from the sum of local choices. This is why, moreover, I prefer the term *self-structured* to the term *unstructured* that is generally used. After all, a structure is, by definition, only *the way in which a set is organized.* For example, with all the properties it possesses (average degree, connected components, diameter, …), the Erdos-Renyi graph, which is built from local decisions (is this edge in the graph?), is for me the perfect example of self-structure.

The self-structured approach, one of whose most representative examples is certainly BitTorrent, is mainly used in content distribution, whether in file sharing or for streaming. It is used in localization in certain systems like Freenet or Gnutella, and can appear in a hybridized form (as mentioned previously, no compartmentalization is completely rigid) in certain DHTs [34, 61].

By construction, self-structured approaches are adapted to the volatility and heterogeneity of peers. But since their structure is not known *a priori*, it is difficult to predict their performance, and validation must often be done empirically. However, advances have been made to give self-structured systems a theoretical foundation [14, 37, 62].

1.3.3 Push and Pull

One can also characterize, from a research perspective, the nature of a peer-to-peer system by looking at where decisions are made: at the client level (*pull*), at the server level (*push*), or a bit of both.

In a pull approach, the content requester will itself initiate the distribution with the content holders[7], possibly deciding which part to retrieve from whom. This is the kind of approach used in the main current file-sharing systems (BitTorrent and eDonkey).

Conversely, the push philosophy dictates that the holders manage the proper functioning of the system, with requesters having a more passive role. Distribution systems based on broadcast trees, where content is routed from the source to the recipients, are generally considered typical of push [19, 35].

As can be seen in the previous examples, this classification is generally associated with the distribution problem. The reason is that the main localization solutions, DHTs (but also centralized localization solutions), do a bit of both: holders proactively transmit the description of their content to a third party, and requesters subsequently go fetch these descriptions. This *publish/subscribe* approach can nevertheless be interpreted, conceptually, as a special case of push/pull hybridization.

Another preconception (which I have intentionally reproduced in the preceding examples) consists of associating push distribution with structured systems and pull distribution with self-structured systems. Intuitively, this association is easy to justify: in a pull approach, each requester organizes itself how it retrieves content, and one might think that a structure too rigid would risk reducing the room for maneuver. Conversely, letting a set of holders manage the distribution of content seems doomed to failure without a minimum of coordination, that is, of structure. And yet, epidemic broadcast algorithms [14], which I will discuss in more detail during Chapter 3 are clearly push-oriented (with a slight pull feedback, let us note), self-structured, and nevertheless destined for what I hope will be a brilliant future.

[7] I recall that in peer-to-peer, one must take requester and holder from a logical standpoint, the holder of one part of content potentially being simultaneously a requester of another part of that same content.

1.3.4 Theoretical, empirical

Beyond the three criteria I have just described, which are related to the specific problems of peer-to-peer, I believe it is also necessary to consider the distinction between theory and practice. This is a classification (even a divide sometimes) that has very distant epistemological origins, but whose repercussions on peer-to-peer are just as important as the distinctions mentioned above. To borrow a definition from Matthieu Latapy, the theoretical approach (or more generally fundamental research) primarily seeks to *understand*, while the goal of the empirical approach is above all to *do* (applied research).

Being somewhat caricatural, the proponent of a purely empirical approach will disparage any research that has not been validated by a series of experiments under real conditions. This is for example the position defended by Bram Cohen, the designer of BitTorrent[8]. Conversely, the purely theoretical approach will seek to propose solutions that can be shown to work, or models that can be thought to capture the essence of a phenomenon. This is how most of the structured solutions cited above proceed, or the fluid modeling of BitTorrent proposed by Qiu and Srikant [63].

In absolute terms, each of the two approaches taken in its pure state is incomplete, and it is only when combined that they can fully express their potential. Either by proposing a theory that allows understanding observations (induction), or by trying to falsify (in the epistemological sense) a theory through a series of experiments. This is unfortunately not always easy, the increasing compartmentalization of domains not really helping to bridge this gap, and a given piece of research (including in peer-to-peer) will always be perceived with a more empirical or more theoretical coloring, the exact nuance depending of course on the eye of the observer.

1.3.5 Summary

As the examples given above have illustrated, the classification I propose is far from being a rigid compartmentalization, and it should rather be seen as a compass allowing one to try to orient oneself within the peer-to-peer research domain. Systems are becoming increasingly hybrid and mix supposedly antagonistic genres. It is no longer possible, and this is not necessarily a bad thing, to completely isolate themes as was the case in an era when the main topics addressed were DHT proposals (localization, structured, publish/subscribe, theoretical) and studies of real traces (localization or distribution depending on the trace, empirical).

[8]This position is perfectly apparent if one consults his blog, notably his post attacking the Avalanche protocol: http://bramcohen.livejournal.com/20140.html.

Chapter 2
Positioning

While it appears clearly, through my work, that my main research theme is peer-to-peer, I am aware that it is not necessarily easy to identify the common thread that unifies my work, even though the concepts of distribution and self-structure frequently appear in the compass quadrant: generic dimensioning of distribution with the bandwidth conservation law [13]; epidemic broadcast with slight delay, that is, self-structured and pull [14]; decentralized video-on-demand distribution (self-structured, with a push phase and a pull phase); self-structure modeling with all of my work on preference networks [54].

In order to better understand my journey, here are the three qualities that I try to bring together as much as possible when I conduct research. I want to emphasize that these are by no means the qualities that define a good researcher in general, because I believe that each researcher has their own profile, their own personality, and therefore their own qualities to develop in order to thrive professionally.

Curiosity I enjoy discovering new themes. Moreover, I particularly appreciate when the subjects I tackle have not yet been studied, or very little. Indeed, this curiosity often pushes me to go beyond or deviate from my initial intentions, and I always fear, when I discover results that I find new and interesting, subsequently discovering that these results have been known for a long time. This is why, after a doctoral thesis whose subject was Google's PageRank, a topic of current interest at the time, I turned quasi-naturally towards the emerging themes of peer-to-peer, but favoring the modeling of distribution, a subject relatively little addressed at the time given its importance.

Listening I am a proponent of scientific discontinuism, in the sense that I do not believe that science evolves according to a continuous, controlled, and quantifiable progression, but rather that it is the sum of abrupt breaks. At a more modest scale, which is mine, I apply this theory by striving to spot the micro-breaks that may come within my reach: I listen and I observe, seizing the opportunity to explore new leads, stemming from my field, other fields, or even my private sphere. The most extreme example is that of the birth of acyclic preference networks, recounted in the foreword, but most of the other themes I have studied are also based on listening to and observing others. Thus, my personal downloading woes in peer-to-peer led me to study the broadcasting mechanisms in file distribution [59]. A few years later, listening to a talk by Laurent Massoulie, I realized that what I had done bore many resemblances to epidemic broadcasting, which pushed me to explore this theme.

Intuition Finally, I am unable to work on a subject without having a strong intuitive understanding of it, either initial or acquired during preliminary work. This is as much a weakness as a quality. It is a quality because intuition makes things simple, and facilitates communicating the essence of one's work to others. It allows one to present one's subject while glossing over the technical difficulties, making it more attractive. It is a weakness when the ideas it gives exceed the possibility of treating them properly[1]. Due to lack of time and adequate technical competence, I thus carry in my closets a certain number of conjectures that I *feel* are true without being able to prove them.

Finally, I would like to discuss the positioning of my research on the slider that goes from theory (understanding) to practice and the empirical (doing). If I had to place myself in absolute terms, which is debatable, I estimate that the research I conduct is positioned primarily at a theoretical level. Personally, and because of my career at the boundary between academia and industry, I prefer to consider myself as an intermediary. This is a pleasant situation when it allows bringing a theoretical perspective to a concrete phenomenon or transforming an equation into a concrete application. It is less pleasant when I realize how weak my individual skills, taken separately, are compared to the leading figures of each *camp*. My programming skills barely allow me to go beyond the simulation stage, and my mathematical toolkit is less well-stocked than that of many researchers I know. But I have decided to turn what I consider a personal weakness into an advantage, and rather than seeking a competition that I am sure to lose in both domains of theory and practice, I prefer to see myself as a bridge between two worlds that ideally should form only one.

2.1 Organization of the thesis

This thesis is structured in three chapters. Chapter 3 focuses on the work I have conducted around the problem of content distribution. It is a synthesis chapter, where I try to present the main ideas of this work, articulated around three sub-themes: the bandwidth law (dimensioning), epidemic broadcasting (simplifying), video-on-demand (decentralizing).

Next comes the heart of this thesis, Chapter 4, which presents my work on acyclic preference networks. In writing this thesis, I had a very strong desire to give a global and unified vision of this domain that I discovered a little over three years ago, which is why I am somewhat more exhaustive here than for a simple synthesis chapter.

[1] "Success is 1% inspiration and 99% perspiration!", *Albert EINSTEIN*

I naturally conclude with a concluding chapter, whose purpose is to place my work in the context of P2P both in the sense in which I have defined it, and in its common understanding. From this synthesis emerge the future research directions that in my view deserve to be studied in order to fully grasp the forthcoming evolutions around the P2P philosophy.

Chapter 3
Content distribution

The purpose of this chapter is to survey the issues related to content distribution in a peer-to-peer approach, focusing on the themes that I have personally addressed during my research. I begin by specifying what I mean exactly by *content distribution* (file sharing, slightly delayed broadcast, video-on-demand), highlighting the specific and common challenges of each type of distribution. I then discuss three specific problems: bandwidth dimensioning (generic to all types of distribution), epidemic broadcasting (slightly delayed), and decentralized video-on-demand

3.1 Distributing

When I speak of distribution, I think first and foremost of large content: computer programs and operating systems, music files, DivX movies, channels broadcast in high definition[1]. For this kind of content requiring large network resources, one can roughly distinguish three ways of performing distribution: file sharing, slightly delayed broadcast, and *on-demand* broadcast.

3.1.1 File sharing

Historically, the flagship application of peer-to-peer in distribution is file sharing, whose success is I think no longer in question. As in the case of a classic (non peer-to-peer) download, it is a *best effort* application by nature, and the effective download time constitutes the main metric. From a network perspective, a challenge is therefore to achieve optimal use of available resources, in order to minimize this download time. Guaranteeing a certain durability of the shared content (avoiding impossible downloads) also plays a role.

[1]The distribution of very small content (for example a peer-to-peer classified ads system), due to the non-existence of bandwidth as a limiting parameter, obeys radically different challenges from those I will present here.

3.1.2 *Streaming* broadcast

A *streaming* application aims to transmit multimedia content for viewing or listening purposes. This kind of application is not at all *best effort*, unlike file sharing, because constraints appear at the time of content playback (mainly, one must ensure that the part of the content one wants to play is there when it is needed). Among the metrics common to all broadcasting, there are:

Quality as a first approximation, it is proportional to the stream bitrate, even though other parameters come into play, such as the choice of encoding or the loss rate.

Initialization delay this is the time that elapses between the moment one decides to watch content and the moment it begins to play.

From a network perspective, the challenge is to

- maximize the useful throughput, notably by minimizing control messages (*overhead*),
- minimize losses,
- minimize the transmission delay (network).

These three parameters are closely related to each other in peer-to-peer. For example, it is generally possible to reduce the transmission delay at the cost of higher losses or overhead. The art of designing a broadcast system then consists of finding the right trade-off.

Slightly delayed

Slightly delayed streaming (the term *live* being somewhat presumptuous on the Internet) seeks to broadcast an event that is currently taking place, for example a football match. In addition to the constraints related to any broadcast over the Internet, the main challenge of slightly delayed broadcast is to minimize the end-to-end transmission delay, that is, the time that elapses between the moment a goal is scored (and therefore when one hears one's neighbors, who are watching the match on television, shout) and the moment one can see it on one's computer screen. From a network perspective, the challenges are the same as for streaming in general, except that the importance of the transmission delay is increased. The very nature of slightly delayed broadcast also creates specific advantages and disadvantages. For example, there is no need to worry about the existence of the content (it is directly injected into the system), but one must be able to satisfy a large number of people watching the same content in a quasi-synchronous manner.

Video-on-demand (VoD)

Video-on-demand essentially consists of combining a file-sharing system with a streaming system. In addition to the constraints related to broadcasting, one must be able to manage a large content catalog, and in particular ensure its availability in an asynchronous manner, each client wanting to watch what they want when they want.

3.2 Dimensioning

The bandwidth conservation law is a tool so simple that many people tend to forget its existence, while it allows, at little cost, to understand and dimension (approximately at least) a distribution system. As I mentioned in the introduction, the first time I became interested in it was when I saw the first discussions about the possibility of sharing files in P2P appear on various forums.

My first reaction was negative: indeed, in a network, if one abstracts away control data, losses, and other multicast replications, data transfer can be compared to a transfer of physical matter: if Alice sends one liter of music to Bob, and if everything goes well, one liter of music leaves Alice's place, flows through the network, and one liter of music ends up at Bob's place. The problem is that the outgoing and incoming pipes do not have the same size in ADSL technology, which is the one used today for most residential connections, and the receiving capacity is generally 4 to 20 times greater than the sending capacity. This is why I was rather skeptical about the future of peer-to-peer: even if all participants fully opened their sending taps, it seemed to me that my bathtub would always fill up more slowly in peer-to-peer than if I used a server capable of saturating my incoming pipe.

This intuition was wrong. Today, when one can download the latest episodes of popular series in less time than it takes to heat up a pizza, I have forged a new intuition, still based solely on the bandwidth conservation law, which allows understanding and predicting (dimensioning) the behavior of a peer-to-peer system.

The key to my error was that I had forgotten that in a peer-to-peer network, not everyone is simultaneously a client and server (that is, a *leecher* to use the standard term). One must account for those who leave their outgoing tap open without using the incoming one (the *seeders*), and also possibly, in the case of hybrid systems, dedicated servers responsible for supporting the system. In the end, the proper way to write the bandwidth conservation law is

$$\sum_{l\in L} d(l) = \min(D_{\max}, U_L + U_S + U_E), \tag{3.1}$$

where L, S, and E respectively denote the population of leechers, seeders, and servers, U_X the total upload capacity of population X, and $D_{\max}$ the total receiving capacity of the leechers.

Despite its simplicity, this equation leads to fairly fine results, the main ones being [13]:

- For a closed *streaming* system there exists a scalability threshold that depends on the bitrate one wants to achieve, the average upload capacity, and the ratio between S and L.
- Below this scalability threshold, a hybrid system allows multiplying the initial server capacity, the amplification depending on the parameters mentioned above.

- For an open system, with arrivals and departures, similar results are obtained by considering the arrival process intensity and the behavior of peers instead of population sizes.
- Unlike a closed system, an open system possesses a zone of over-performance, whose boundary is defined by the behavior of seeders, where leechers can receive at maximum capacity regardless of the arrival intensity. Intuitively, an open system can accumulate enough seeders to make the transfer capacity arbitrarily large. An enforced sharing ratio policy is a good way to approach this critical zone.
- For tit-for-tat incentive systems, a la BitTorrent, there exists a tolerance threshold for selfish users (free-riders). Beyond this threshold, the system can no longer expel free-riders, who accumulate continuously in the system.

3.3 Simplifying

When slightly delayed broadcast became the objective to achieve in peer-to-peer, the theoretical and practical currents followed two radically different paths: on one side, theoretical solutions were proposed based on trees with disjoint interior nodes [19, 35]. The problem with these techniques is that for various reasons that I will not have space to discuss here, there currently exists no viable client based on them. As a consequence, these structured approaches have not yet managed to reach the general public.

On the other side, the practical camp started from what already worked, namely file-sharing applications like BitTorrent, and sought to adapt it to broadcasting. This resulted in a generation of new applications like PPLive [41], which enjoy growing success with the general public. But the problem with this approach, even though the result is functional, is that BitTorrent is not designed for live broadcasting. In particular, while a live stream should be transmitted as linearly as possible, the ideal being to receive the stream pieces in the order in which they were created, BitTorrent is designed to shuffle the order of pieces as much as possible in order to maximize availability and avoid the appearance of a *missing piece* [59]. To work around this contradiction, the solution generally employed is to group successive pieces into macro-pieces. Each macro-piece taken individually is broadcast *a la BitTorrent*, while the retrieval of successive macro-pieces is done linearly, which ultimately allows reading the stream in the right order, macro-piece after macro-piece. The drawback is that in the end, the performance obtained, notably in terms of initialization delay and end-to-end transmission delay, falls well below the promises of theory.

Faced with this alternative, a third way, which has held my attention during my work, is that of epidemic broadcasting. If I had to give a single reason to justify my choice, it would be their simplicity (not to be confused with triviality). Indeed, while structured algorithms require setting up a fairly complex structure and current applications add an additional layer to an existing algorithm that is already quite elaborate, the design of an epidemic broadcast is much simpler. When a peer has the technical ability to send a piece of the stream to one of its neighbors, it must answer the following two questions:

- *To whom should it send?*
- *Which piece should it choose to send?*

Each way of answering these two questions (which one will notice are very close to the *Who retrieves what from whom?* that I proposed to define peer-to-peer) corresponds to an epidemic broadcast algorithm. One can thus very easily generate a quasi-infinite number of distinct algorithms, and evaluate their performance through theoretical analysis or simulations [14].

Without going into the technical details, I would say about the ongoing research that a consensus seems to be emerging regarding the choice of piece (second question): once the recipient is chosen, the most efficient approach seems to be to send the *latest useful chunk*, that is, the most recent piece that the target does not yet possess. Although variations are still under study, this *latest useful* strategy currently gives the best performance in terms of delay.

For the first question, the choice of recipient, the question is still far from settled: we know that in certain cases, choosing the recipient simply at random gives quasi-optimal performance on a theoretical level, but simulations suggest that there is much room in that *quasi*. Current research (conducted at Orange Labs, but also by other teams in France and in Europe) therefore aims to do better than random selection, knowing that to choose otherwise than at random, the sender must be given more information, and that there is a trade-off to find between the amount of information to be provided to the sender at the time of its choice and the quality of the resulting choice. In the extreme, one could imagine a mechanism where senders communicate in order to synchronize and manage to create an optimal broadcast, but the cost of such a technique in terms of control messages (*overhead*) would very likely be prohibitive.

3.4 Decentralizing

Finally, I would like to close this chapter on content distribution by briefly discussing the problem of decentralized video-on-demand, which I had the opportunity to address under the impetus of Laurent Viennot [15, 16]. This problem, originally proposed by Suh *et al.* [74], uses the fact that more and more households are equipped with *set-top boxes*, which among other features have a hard drive and Internet access. Hence this very simple idea: since a video-on-demand service essentially requires storage capacity and bandwidth to operate, why not build a service that relies on the resources of the *set-top boxes*?

The interest of this subject is that beyond the classic problems encountered in distribution (bandwidth and initialization delay for example), we see the emergence of a trade-off between bandwidth, storage capacity, and failure rate. Thus, if video files are stored with low redundancy, the closer the available bandwidth is to the limit necessary for the proper functioning of the service (conservation law), the greater the risk that in practice, when it comes time to distribute a video, the set-top boxes possessing that video have already exhausted the necessary network resources, leading to distribution failure. To reduce the risk of failure, one must either increase network capacities or increase redundancy, and therefore mechanically decrease the available storage capacity[2].

Another specificity of this problem is that although very simple practical solutions, based on random choices, can be implemented to solve it in a nearly optimal manner, the underlying allocation theory is very subtle and assuredly non-trivial.

[2]The way in which videos are segmented also plays a role [15, 16].

Chapter 4
Acyclic preference networks

The objective of this chapter is to present a contribution that is particularly dear to me, namely the *preference networks* model. I proposed this model in order to better understand the dynamics of collaborations between peers who act independently according to their own preferences.

This chapter is structured in four parts: I first propose to give a brief overview of the theory of stable marriages, from which the preference networks model is derived. I then introduce the main foundations: the formalism employed, the grand theorem of acyclic preferences, which is at the center of the theory, and finally a taxonomy of these famous acyclic preferences. The third part is then devoted to a detailed study of the self-stabilization properties revealed by the grand theorem, while the last part is devoted to the methods for describing the stable configuration of an acyclic preference network.

4.1 Origin

The origin of preference networks is the theory of *stable marriages* proposed by Gale and Shapley. The most entertaining version of this problem appears in the original 1962 article [38]: it consists of marrying men and women while ensuring that any risk of adultery is avoided. Formally, an instance of the stable marriage problem consists of two sets H and F (the men and the women), each strictly ranking a subset of the members of the opposite sex. A configuration of this instance is simply a matching between H and F. It is unstable if there exists a pair (h, f) not belonging to the configuration, such that h prefers f to his partner in the configuration (or if h is single, that h accepts being with f), and reciprocally. If this occurs, adultery is not far off, because h and f both have an interest in getting together, even if it means the end of their current couple. Starting from this definition, the objective of the theory is to identify the stable configurations of a given instance. It was proved as early as 1962 that if preferences are strict (no ties in the rankings), there always exists a stable configuration, which can be easily obtained by the *proposal algorithm* [38].

A first extension of the problem is that of *college admission* (*college admission* or *hospitals/residents* in the original English terminology [43, 66]). In this variant, members of one of the sets can choose several representatives from the opposite set. For example, a university can open several faculty positions each year, but a future faculty member will go to only one university. As with the stable marriage problem, it is always possible to obtain a stable configuration, and certain algorithms proposed by Gale and Shapley are still in use in many recruitment processes.

But for P2P, the interesting extension is the one where there is only a single category of participants. This unisex variant is known as the *roommates* problem in the case of simple matching, and b-matching in the general case, b being the standard notation for the vector describing the quotas of the multi-matching [31, 42, 44]. Unisex matching finds applications in numerous domains, one of the most active variants being pairwise kidney exchange programs [67, 80]. The main difficulty compared to sex-typed marriages is that it is no longer possible to guarantee the existence of a stable configuration, which can prove problematic, particularly when it comes to organ donation.

4.1.1 Reinventing the wheel?

Intuitively, the interest of marriage theories for P2P is quite obvious: if one has a set of peers, each possessing a list of neighbors with whom it is likely to interact, the theory should be able to shed light on what happens assuming that each peer acts according to its own interest (i.e., its own preferences).

Since marriage theories have existed for nearly 50 years, one might think that it suffices to take the existing literature to have the ideal P2P modeling tool. In practice, things are not so simple: marriage theories do indeed offer an extremely powerful theoretical foundation, but the specificities related to P2P require seeing the problem from a new angle, that of preference networks, which are the subject of this chapter.

More precisely, the main changes brought by preference networks are the following:

- the preferences of classical b-matching problems are often arbitrary, whereas in P2P, most preferences are derived from objective measurements;
- the existence of stable configurations and the means to obtain them are an essential component of b-matching problems; conversely, most P2P preferences benefit from a powerful existence/uniqueness/convergence theorem that makes these questions moot;
- the questions of dynamics and performance of configurations are a fundamental aspect for P2P, and must therefore be highlighted.

4.2 Foundations

The purpose of this section is to present the field of preference-based networks, in broad strokes, paved and marked out after three years of exploration: the notations and definitions, as they have stabilized over time (Section 4.2.1); the grand theorem of acyclic preferences, which despite its simplicity is the backbone of the model (Section 4.2.2); and finally, the taxonomy of acyclic preferences, which shows how to relate the model to real-world situations (Section 4.2.3).

4.2.1 Notations and definitions

As mentioned in the introduction, preference-based networks originate from the theory of b-matchings [20, 24], and the notation reflects this, even though a specific sub-vocabulary has been introduced over the course of various works [36, 37, 50, 55, 56]. Regarding the French version in particular, the often colorful vocabulary owes much to Dominique Dumont's popular science article *Mariages Stables* (Stable Marriages) [28].

Thus, an instance of a preference-based network consists of a set of peers, an acceptability graph, preferences, and quotas. A configuration is a set of interactions, whose dynamics are described by peer initiatives.

Preference-based network

The preference-based network itself thus consists of a set P of n peers (or nodes) and a graph $G = (P, E)$. Preferences are indicated by a value matrix m and quotas by a vector b.

The acceptability graph $G = (P, E)$ is an undirected and non-reflexive graph. It describes compatibilities: two peers i and j can interact if, and only if, (iff) $\{i, j\} \in E$. We then say that i is *acceptable* for j, and vice versa. For example, G can represent knowledge of other peers (it is rare for a peer to know all other peers in a network; its view is generally limited to a certain number of *neighbors*), or the existence of a common interest (searching for the same file, belonging to the same group, ...), or simply an imposed *overlay* structure. The graph G can in principle be arbitrary; however, we will often use in this chapter Erdős-Rényi graphs $\mathcal{G}(n, p)$, where each edge exists with probability p independently of the others (the average degree is thus $d = p(n-1)$). This choice is justified by the facilities that $\mathcal{G}(n, p)$ graphs offer for the theoretical analysis of preference-based network properties.

The value matrix m indicates the interest that peers have in one another. It thus determines preferences. For every acceptable pair of peers $\{i, j\} \in E$, $m(i, j)$ is the value that i assigns to j. Unless otherwise stated, lower values are preferred[1].

[1]This choice is purely conventional and does not affect any of the results presented here. It is natural when considering latencies or distances. In the case of values such as bandwidth, or more generally capacities, the opposite would be more appropriate.

Thus, $m(i, j) < m(i, k)$ means that i prefers j to k. An assumption made throughout this chapter is that there are never ties, that is, for each row of m, the acceptable values are pairwise distinct[2]. The matrix m may possess certain special properties that will be described in Section 4.2.3. I also note that while it is always possible, and convenient, to assume the existence of a value matrix m, this is not essential from a theoretical standpoint, since only the orders (preferences) induced by m matter [10, 36]. It is not necessary for the peers to know m. In the worst case, a peer may not even know the values in its own row. However, in order to preserve the meaning of the value matrix concept, we assume that it is always possible for a peer to compare the values of two neighbors by contacting them, for example during an *initiative* (see below).

Finally, the quota vector, b, limits the number of collaborations: a peer i cannot collaborate with more than $b(i)$ neighbors simultaneously. One can of course always assume that $b(i)$ is no larger than the degree of i in G. Just as G is often assumed to be Erdős-Rényi, b is generally assumed to be constant when analyzing a network, even though any distribution is allowed in principle.

Configurations

The configuration of a system describes the state of collaborations. Formally, a configuration C is a subset of E. The neighbors $C(i)$ of i in C are the current *partners* of i. If $c(i)$ denotes the number of partners of i in C, the quotas imply $c(i) \leq b(i)$. If $c(i) < b(i)$, we say that i is *undermatched*.

A configuration can evolve through the resolution of *adulterous* edges (the term is a tribute to the original theory of stable marriages). An acceptable edge e is adulterous for a configuration C if it does not belong to C and if each peer of the edge has an interest in establishing the collaboration e, even if this means abandoning a collaboration in C to satisfy the quotas. Formally, an edge $e = \{i, j\} \in E \setminus C$ is adulterous iff:

- $c(i) < b(i)$ (i is undermatched), or $\exists k \in C(i), m(i, j) < m(i, k)$ (i prefers j to one of its partners);
- $c(j) < b(j)$ or $\exists k \in C(j), m(j, i) < m(j, k)$ (symmetric condition).

A peer that is adjacent to at least one adulterous edge is *eligible* (it is likely to modify the configuration). A configuration that has no adulterous edge (or equivalently no eligible peer) is *stable*.

Initiatives

Adulterous edges are at the basis of configuration dynamics. More precisely, evolution always comes from an eligible peer that tries to resolve an adulterous edge to which it belongs. This process, assumed to be atomic, is called an *initiative*. An initiative is *active* if it results in the resolution of an adulterous edge.

[2]Ties in preferences are a source of problems that I do not wish to discuss in this chapter [43–45, 52, 53, 65]. We will therefore assume that ties can always be broken, for example by assigning each peer a unique identifier to resolve any ambiguity.

Formally, the initiative is a selection "function" (which may depend on time, the system state, be random, ...) that assigns to each peer i in V a neighbor to "try", or more formally an edge of $E \setminus C$ incident to i. While by default, all such edges may be considered by this selection function, in the case where peers have knowledge of the adulterous edges to which they belong, it is possible to restrict the selection to these edges only.

Examples of initiatives include *best partner* selection (among adulterous edges), *decremental* selection (round-robin choice among the list of neighbors), or simply random selection. Note that the different types of selection implicitly require more or less knowledge: for instance, *best partner* and *decremental* require the ability to sort one's neighbors (and even to identify adulterous edges for the former), whereas a purely *random* selection only requires the ability to evaluate a neighbor on the fly. Finally, it is always possible to hybridize several selections to obtain a more efficient one. An example of a *hybrid* selection will be described in Section 4.3.2 (after [55]).

Starting from a given initial configuration, the evolution of a system is thus described by the sequence of initiatives performed by the peers. A classical sequence is the *Round-Robin* sequence, suitable for modeling single-period peer behavior, or the *uniformly random* sequence, used to model a homogeneous Poisson process (each peer follows an i.i.d. Poisson process). Finally, there are *adversarial* sequences, which try to behave in the "worst" possible way.

For convenience, time is measured directly in terms of the sequence of initiatives (there is thus no need to explicitly introduce temporal initiative processes). It is nevertheless possible to consider several measures depending on the context. For example, in the context of self-stabilization in Dijkstra's sense [25], only active initiatives are counted, and sequences are divided into *rounds* (a round is a subsequence such that each peer eligible at the beginning of the round takes an initiative or becomes non-eligible during that round). For simulations, I prefer to define the time unit (t.u.) as n initiatives (active or not). Thus, each atomic initiative takes $\frac{1}{n}$ t.u., and after t t.u., the number of initiatives per peer is on average t. When the initiative sequence is round-robin, round and time unit can be considered synonymous, but this is not true in the general case.

Local stability

The notion of initiative leads to an alternative definition of stability: a configuration C is stable if the only configuration reachable by initiatives from C is C itself. It is easy to verify that for arbitrary initiatives, this definition is equivalent to the one given in Section 4.2.1.2 (no adulterous edge). The interest of an initiative-based approach to stability is that it allows a local definition: an edge e of a configuration C is *stable* iff e exists in all configurations reachable from C. In other words, a stable edge is a collaboration that initiatives cannot break.

When a peer is incident to a stable edge, its degree of freedom is reduced. The free quota of a peer i, denoted $b'(i)$, is the quota $b(i)$ minus the number of stable edges incident to i. The free quotas form a vector that is a function of the configuration and that only decreases over time. It is thus possible to extend the definition of stability to the peer level: a peer i is *stable* (or *deactivated*) if $b'(i) = 0$ or if i shares a stable edge with all its non-stable neighbors.

Finally, a last fundamental notion regarding stability is that of *heat*. In a given configuration C, an acceptable edge $e = \{i, j\}$ outside of C ($e \in E \setminus C$) is hot iff:

- j belongs to the $b'(i)$ best non-stable neighbors of i,
- i belongs to the $b'(j)$ best non-stable neighbors of j.

There also exists a stricter definition of heat: an acceptable edge $e = \{i, j\}$ between two non-stable peers that are not collaborating is said to be boiling for C iff i is the best non-stable neighbor of j and vice versa.

By extension, a peer is said to be hot (resp. boiling) if it is adjacent to a hot (resp. boiling) edge.

Intuitively, hot edges, and even more so boiling ones, are super-adulterous edges that only need a well-placed initiative to become stable edges. They play a central role in acyclic preference-based networks, because of Lemma 4.1, which is used in many proofs:

Lemma 4.1 ([36, 50]): If C is a non-stable configuration of an acyclic preference-based network, then there exists at least one boiling edge with respect to C.

Obviously, this lemma only makes sense once the notion of acyclicity has been defined. This is in fact the subject of the remainder of this section on the foundations of preference-based networks.

4.2.2 Grand theorem of acyclic preferences

A *preference cycle*, or *Kieschnick cycle*, is a cycle of $k \geq 3$ peers $i_1, ..., i_k$ such that each peer prefers its successor to its predecessor: i_1 prefers i_2 to i_k, i_2 prefers i_3 to i_1, ..., i_k prefers i_1 to i_{k-1} (or, expressed in terms of values, $m(i_1, i_2) < m(i_1, i_k)$, $m(i_2, i_3) < m(i_2, i_1)$, ..., $m(i_k, i_1) < m(i_k, i_{k-1})$).

The main subject of this chapter is acyclic preference-based networks, that is, networks that do not contain any Kieschnick cycle. Note that acyclicity is entirely defined by m, and depends neither on the quotas nor on the initiatives. Similarly, if m produces acyclic preferences for complete acceptability, then the network will be acyclic for any acceptability graph G. By convention, such a value matrix m is also called acyclic. A classification of acyclic matrices, and their connection to P2P systems, are proposed in Section 4.2.3.

Acyclic preference-based networks are characterized by a property that makes them unique among preference-based networks:

Theorem 4.1 ([36]): An acyclic preference-based network admits one, and only one, stable configuration. Moreover, it is self-stabilizing by initiatives.

Proof: The complete proof is available in [36], but I give the idea here, because of the importance of this theorem and the typical nature of the techniques employed. We proceed in two steps: first show self-stabilization (which will prove the existence of a stable configuration), then uniqueness.

Self-stabilization comes from the following property: regardless of the initiative sequence considered and the starting configuration, the corresponding sequence of configurations is irreversible. That is, if the system was in a configuration C_1 in the past and is now in a configuration $C_2 \neq C_1$, then C_1 is not reachable from C_2. Irreversibility is proved by contradiction, by extracting a Kieschnick cycle from a cycle of configurations. Since n is assumed finite, the number of possible configurations is also finite, which means that any trajectory of configurations necessarily leads to a configuration that no longer evolves, i.e., a stable configuration.

Uniqueness is also proved by contradiction: if one considers two distinct stable configurations A and B of the same system, and a peer i whose collaborations differ from A to B, then one can construct a Kieschnick cycle starting from i (even if i does not necessarily belong to the cycle ultimately obtained).■

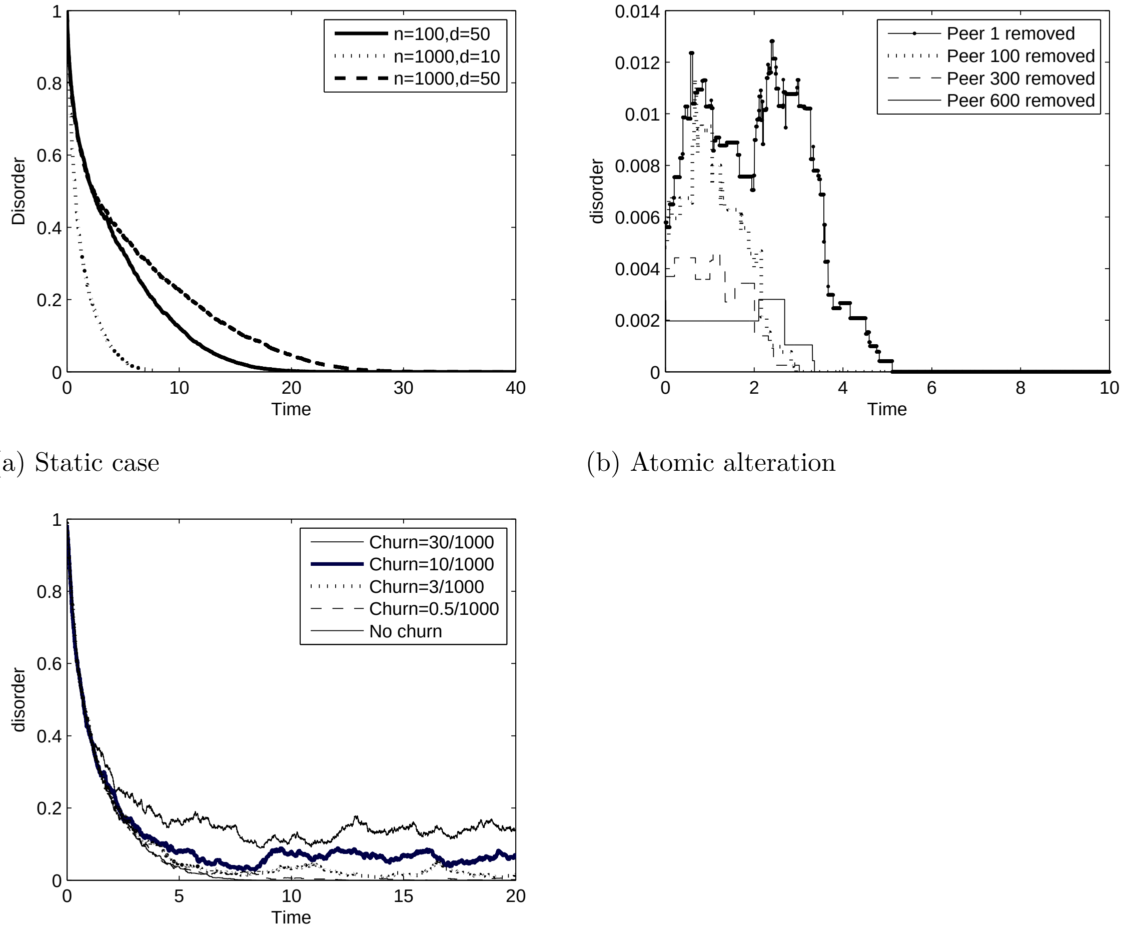
I emphasize once again the fundamental nature of this existence/uniqueness/self-stabilization theorem, as it allows us to dispense with the stability analysis performed for other types of preferences, and to focus more closely on the configurations themselves.

The first application of this theorem appeared in the article *Stratification[3] in P2P Networks: Application to BitTorrent* [37]. In this article, which sought to model a BitTorrent network as an acyclic preference-based network (I will discuss this in more detail later in the chapter), we measured the effective convergence between actual and stable configurations. To estimate the practical importance of self-stabilization, we had considered three types of scenarios (cf Figure 4.1):

Static (Figure 4.1a) Starting from the empty configuration (often denoted $C_\emptyset$), we observed convergence toward the stable configuration. We observed good convergence in all cases, while noting that the system parameters, and the average degree of the acceptability graph in particular, played a major role;

Atomic alteration (Figure 4.1b) A second elementary scenario consists of removing a peer from a stable configuration (which modifies the stable configuration), and letting the system converge again. We observe that the actual

[3] Many people ask me why *stratification*, instead of *clustering* for example. Well, first of all, because *clustering* is already semantically overloaded in computer science; then because the image of strata is at least as evocative; and finally because the word is elegant.



(a) Static case

(b) Atomic alteration

(c) Continuous churn

Figure 4.1: Self-stabilization in practice (after [37]).

configuration does not deviate much from the stable configuration, and eventually re-converges, even though, due to a possible domino effect, the convergence time may be comparable to the static case.

Continuous churn (Figure 4.1c) Finally, a third scenario consists of having peers join and leave the system at a certain rate. This scenario shows that the actual configuration does not always manage to catch up with the evolution of the stable configuration, especially when churn is extreme. However, self-stabilization ensures that the actual configuration remains reasonably close to the stable configuration, the *distance* between the two being roughly proportional to the churn.

Following this study, I began working under the assumption that, provided the system's convergence is sufficiently fast compared to its evolution, the stable configuration should be a good approximation of actual configurations. This is the *spring* metaphor (cf Figure 4.2). This metaphor suggests decomposing the study of acyclic preference-based systems into two distinct problems:

What is the convergence speed? Indeed, Theorem 4.1 only gives us the number of possible configurations as a bound. This number grows factorially [21] and thus

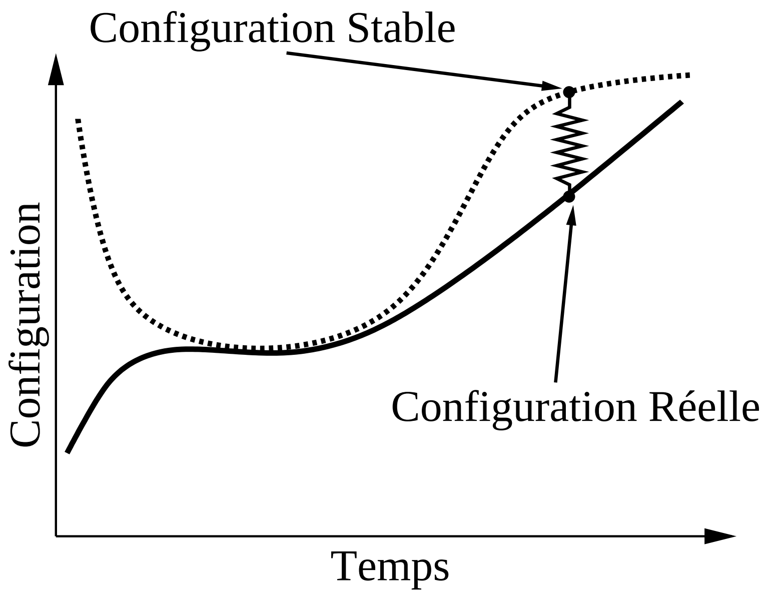


Figure 4.2: The spring metaphor: self-stabilization can be seen as a continuous attraction of the current configuration toward the stable configuration, as if a spring connected these two configurations. This principle holds even if the stable configuration moves over time.

offers rather limited practical interest. A first area of study therefore consists of characterizing more precisely the effective *stiffness* of the spring.

What are the properties of stable configurations? The previous question should help determine to what extent a stable configuration is a good approximation of an actual configuration. When this is the case, the generic properties of stable solutions can help estimate system performance.

These two questions will be the subject of the next two sections of this chapter. But before answering them, I propose to take a closer look at these acyclic preferences, of which I have so far given only a summary and impractical definition.

4.2.3 Taxonomy of acyclic preferences

Among all possible real matrices, few are acyclic. In fact, if one takes a matrix whose coefficients are chosen at random, it is very likely that the resulting preferences contain at least one cycle of length 3[4]. On the other hand, many matrix properties are synonymous with acyclicity. For example ([36]):

Global matrices A value matrix m is global if all its rows are identical when restricted to acceptable entries. Intuitively, this is a special case of a rank-1 matrix, and formally, $m(i,k) = m(j,k)$ for all i, j that accept k. The corresponding preferences, also called global, reflect a total order on the peers. One

[4]The probability that three mutually acceptable peers form a cycle is $\frac{1}{4}$, for uniform random values. The probability of having no such cycle is therefore at most $\left(\frac{3}{4}\right)^{\binom{n}{3}}$ (if one neglects possible correlations), which is not worth much as soon as n is large enough.

can moreover note that because of this total order, up to acceptability and permutation of peers, there is only one global preference.

Symmetric matrices If m is symmetric on its acceptable entries ($m(i,j) = m(j,i)$ for every acceptable pair), m is acyclic. An interesting property of symmetric matrices is that they describe the set of all acyclic preferences (cf [10, 36]). The relationship between acyclic, symmetric, and global matrices, and the corresponding preferences, is illustrated in Figure 4.3[5].

Linear combination of global and symmetric matrices The resulting matrices are acyclic as long as they do not generate ties (but as noted in Section 4.2.1.1, it is always possible to eliminate ties). On the other hand, a linear combination of acyclic matrices is not necessarily acyclic: if the input matrices are not themselves already linear combinations of symmetric and global matrices, they must be symmetrized, which is fairly easy since the proof that symmetric matrices surject onto acyclic preferences is constructive (cf [10, 36]).

Preferences suited to P2P

Having outlined what makes preferences acyclic, I now propose to refocus on the peer-to-peer context and consider acyclic peer-to-peer preferences.

Capacities A peer in a P2P network possesses many intrinsic scalar characteristics that can create preferences: access bandwidth, storage or computing capacity, average *uptime*, … Taking the example of the file-sharing protocol BitTorrent [23], a *tit-for-tat* algorithm causes a peer to preferentially collaborate with neighbors that provide it the highest download speed. This can be seen, as a first approximation, as a preference-based network where the value is the upload bandwidth divided by the quota (the number of simultaneous uploads).

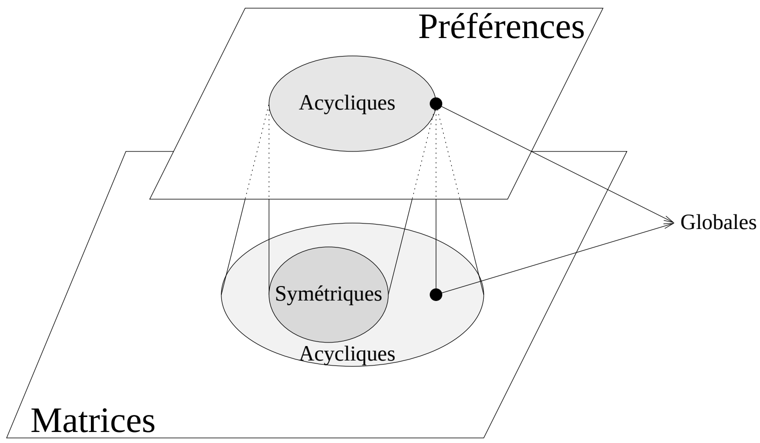


Figure 4.3: Correspondences between matrices and preferences: symmetric and global matrices form two distinct subsets of acyclic matrices, but symmetric matrices can describe the entire set of acyclic preferences (which includes global preferences).

[5]Figure 4.3 is of course incomplete. It is missing, for example, linear combinations in general, and complementary matrices in particular, as well as the generation of the set of all acyclic preferences by permutations of the entries of a symmetric matrix with distinct coefficients. But graphical readability would suffer.

Proximities All values that can be assimilated to some kind of distance (physical or virtual) or similarity are symmetric by nature. Thus, many P2P systems try to minimize latencies[6], such as the DHT Pastry [68] or real-time online gaming applications [51]. Similarly, massively multiplayer online games (MMOGs) must connect players that are close to each other in some virtual space [33, 47, 48]. Some authors also propose connecting participants of a file-sharing system based on their common interests [29, 71], which remains a symmetric measure. *Co-uptime*, that is, the (average) common active time, is a last example of a symmetric measure of interest for collaborative applications.

Complementarities Measuring differences between participants' resources can also prove interesting. For example, in a distributed file storage application, it is useful to find machines that are on when my own machine is off, in order to keep my data available at all times. The corresponding measure is complementary uptime. Similarly, in a system like BitTorrent, all participants seek to have the same file, which is divided into blocks[7]. Finding neighbors that have the most blocks I have not yet obtained is of interest. All these so-called complementary preferences are a special case of a linear combination of a global matrix and a symmetric matrix, and are therefore acyclic [36].

Preference classes studied

As we have just seen, there exist many acyclic preferences of interest from the P2P perspective. Over the course of my work, I focused on four classes of acyclic preferences, which avoids getting lost in a maze of special cases while giving a precise idea of the links between preferences and system properties.

Global Although representing a negligible subset (in terms of cardinality) of acyclic preferences, global preferences are among the most important acyclic preferences (they model capacities), and must receive particular attention. Since there is in fact only one unique global preference system, induced by a total order on the peers, I will use this order instead of making the value matrix m explicit. Peers will thus be labeled from 1 to n, 1 being the *best* (it is preferred by all its acceptable neighbors), and so on…

Geometric For preferences derived from distances, the properties of preferences depend greatly on how the points are positioned in the underlying space. One can moreover easily show that all acyclic preferences can be generated by placing the n peers at the vertices of a simplex and perturbing it (by slightly moving the vertices) in an appropriate manner. Nevertheless, experience seems to show that certain characteristics occur frequently. To obtain typical and analyzable proximity preferences, I propose to consider preferences obtained by taking distances between n random points on a unit t-torus of dimension t ($t \geq 1$).

[6] The question of whether latencies are distances, or even symmetric, naturally arises. In any case, empirically, they do not produce Kieschnick cycles.

[7] It even appears that sometimes, some blocks may be rarer than others [59].

Real latencies Having real measurements to validate results is obviously necessary, even though theoretical analysis then becomes difficult. For $n \leq 2500$, I use, to check the validity of certain geometric results, subsets of size n extracted from the Meridian project dataset [6]. The values are the (symmetric) latencies between the selected nodes.

Random acyclic A last class of preferences is obtained by assigning to each acceptable edge a uniform random value between 0 and 1. I call the resulting preferences *random acyclic* (or simply acyclic when there is no ambiguity), because it can be shown that this approach produces a uniform sampling over the set of all possible acyclic preferences [10, 36].

4.3 Characterization of self-stabilization

What is the speed of convergence? This section will attempt to answer this question, from both a theoretical (Section 4.3.1 and Section 4.3.2) and a practical standpoint.

4.3.1 Upper bounds

I first propose to give upper bounds, in the sense of self-stabilization theory. The initiative considered throughout this section is *best partner*. For $b = 1$, the bounds are exact, linear in rounds, and quasi-exact, exponential in initiatives. I then give weaker bounds for $b > 1$.

Convergence (rounds)

For $b = 1$, the maximum convergence time, expressed in rounds, is given by the following theorem.

Theorem 4.2 ([56]): Starting from any configuration, an adversarial sequence converges in at most $\lfloor \frac{n}{2} \rfloor$ rounds. This bound is achieved by a round-robin sequence.

Proof: The Lemma 4.1 guarantees the stabilization of at least one boiling edge per round. Since there are at most $\lfloor \frac{n}{2} \rfloor$ edges to stabilize, the bound follows immediately. To achieve it, one can use global preferences, a complete acceptability graph, the initial configuration $C_\emptyset$, and a round-robin sequence using the *worst to best* pattern. ■

Convergence (initiatives)

If we now want to measure convergence in number of active initiatives, the bound depends on the constraints of the sequence: round-robin or fully adversarial.

Round-Robin Sequence

The bound for round-robin sequences is given by the following theorem.

Theorem 4.3 ([56]): Starting from any configuration, a round-robin sequence converges in at most $\sum_{k=0}^{\lfloor \frac{n}{2} \rfloor - 1} n - (2k+1) \approx \frac{n^2}{4}$ active initiatives. This bound is exact.

Proof: As for Theorem 4.2, we use the fact that at least one boiling edge is stabilized per round. Moreover, a stable peer is by definition never active again; the precise counting of possible initiatives then gives $\sum_{k=0}^{\lfloor \frac{n}{2} \rfloor - 1} n - (2k+1) \approx \frac{n^2}{4}$. The system used for Theorem 4.2 achieves the bound, proving that it is exact.■

Adversarial Sequence

Convergence for an adversarial sequence is described by the following theorem.

Theorem 4.4 ([56]): Starting from any configuration, an adversarial sequence converges in at most $2^{n-1} - 1$ active initiatives. Conversely, there exists a sequence that converges in $\Theta(\lambda^n)$ active initiatives, with $\lambda \approx 1.6826$, which shows that the exact bound for adversarial sequences lies between these two values.

Proof: Like many other proofs, the bound $2^{n-1} - 1$ follows from Lemma 4.1. The trick is to consider a boiling edge and look at the moment it stabilizes. Before that instant, the adjacent peers cannot take the initiative: we are by hypothesis in *best partner* mode, so they would stabilize. Afterwards, they are inactive. We deduce a recursion on the number of peers allowed to take an initiative, from which the bound is derived.

The bound $\Theta(\lambda^n)$ is obtained for global preferences, with complete acceptability, starting from a *worst first* sequence: at each instant, the initiative is taken by the worst active peer (the one with the largest label). A complete study of the behavior of this sequence of initiatives gives the lower bound. ■

Remark 4.1: In all the results I have just stated, the bounds are achieved from global preferences. From this point of view, one can therefore consider that global preferences are the “worst” possible for convergence, as the sequel will confirm. More generally, global preferences often have atypical behavior among acyclic preferences.

Generalization to b-matching

For $b > 1$, an adaptation of Theorem 4.2 immediately gives us a bound of $\lfloor n\frac{b}{2} \rfloor$ rounds (at least one edge is stabilized per round). But it is easy to see that this is just an upper bound. For example, taking $b = n - 1$ (this is the limiting case of the absence of quotas: everyone can simultaneously collaborate with everyone), each peer is guaranteed to be stabilized after taking $n - 1$ initiatives, giving a bound of $n - 1$ rounds much smaller than $\lfloor \frac{n(n-1)}{2} \rfloor$.

The explanation behind the imprecision of the bound is that when $b > 1$, a non-stable configuration has hot edges in addition to boiling edges. These merely hot edges are harder to enumerate (not all transient configurations have them), which makes computing the bounds more difficult. We can still give bounds more precise than $\lfloor n\frac{b}{2} \rfloor$ for global preferences (which, let us recall, are intuitively the worst possible preferences).

Theorem 4.5 ([55]): For global preferences, the convergence time is bounded by $\frac{b}{b+1}n$ rounds if the acceptability is complete, and by n otherwise.

Simulations (cf Section 4.3.3) indicate that the bound $\frac{b}{b+1}n$ reflects fairly well the actual behavior of global preferences: the first quotas are the most expensive in terms of convergence. Complete acceptability also seems to be the worst case, which means that n is clearly an overestimate (not to mention $\lfloor n\frac{b}{2} \rfloor$).

Proof: If the acceptability is complete, the stable configuration consists of cliques of size $b + 1$ ([37]); b rounds stabilize a clique, hence the result. In the general case (unknown acceptability), we can only assert that b rounds stabilize b peers, hence the bound n. ∎

4.3.2 Average convergence times

Let us now turn to the average convergence for non-adversarial round-robin and Poisson sequences. The results presented here hold for $b = 1$, with the *best partner* initiative. These are upper bounds on the average convergence time of certain classes of preference systems, where average time is measured in initiatives (1 u.t. = n initiatives).

Generic bound

Under the assumptions considered ($b = 1$ and *best partner* initiative), the following bound holds for any acyclic system:

Theorem 4.6 ([55]): The average convergence time is bounded by $\frac{n}{4}$ for a Poisson sequence, and by $\frac{n+1}{6}$ for a round-robin sequence.

Proof: As often, the theorem relies on Lemma 4.1, namely the existence of at least two boiling peers in any non-stable configuration. We deduce an average time between two "boiling" initiatives of $\frac{1}{2}$ for a Poissonian sequence and $\frac{1}{3}(1+\frac{1}{n})$ for a round-robin sequence. Multiplying by $\frac{n}{2}$ (maximum number of edges in the stable configuration), we obtain the result. ■

Remark 4.2: For global preferences with complete acceptability, any unstable configuration admits exactly two boiling peers (the two best non-stabilized peers). This suggests that the bounds of Theorem 4.6 should be fairly precise for these systems, which is confirmed by the simulations [55]. We recover the fact that global preferences are "the worst possible acyclic preferences".

Remark 4.3: The exact bound $\frac{n+1}{6}$ for round-robin sequences is not contradictory with the exactness of the bound $\lfloor\frac{n}{2}\rfloor$ of Theorem 4.2: one is an average bound over all round-robin sequences, the other a worst-case bound.

Global preferences

For global preferences, with an Erdős-Rényi acceptability graph $\mathcal{G}(n,p)$, there exist finer bounds:

Theorem 4.7 ([55]): With global preferences, if $G=\mathcal{G}(n,p)$, the average degree being $d=p(n-1)$, the average convergence time is $O(d+\log(n))$ for a Poissonian sequence and $O(d)$ for a round-robin sequence.

Proof: The core of the proof consists in *counting* the boiling peers in non-stable configurations. Using combinatorial techniques, one shows that there are of the order of $\frac{n}{d}$. We then deduce an average time between two "boiling" initiatives of the order of $\frac{d}{n}$, which gives the common behavior in $O(d)$. Finally, particular attention must be paid to the finalization of the convergence process, when almost all edges are already stabilized. In this *endgame*, the non-stable nodes are almost all boiling. A round-robin sequence will then complete the stabilization in $O(1)$ (1 time unit stabilizes all currently boiling nodes), while a Poissonian sequence will need $O(\log(n))$ t.u. (balls and bins problem). ■

Acyclic preferences

For (random) acyclic preferences, we have a similar result:

Theorem 4.8 ([55]): For random acyclic preferences, if $G = \mathcal{G}(n, p)$, the average degree being $d = p(n-1)$, the average convergence time is $O(\log(d) + \log(n))$ for a Poissonian sequence and $O(\log(d))$ for a round-robin sequence.

Proof: A calculation can show that for random acyclic preferences, approximately half of the non-stable peers are boiling. This allows showing that the endgame is reached after $O(\log(d))$ time units. If the sequence is Poisson, an additional $O(\log(n))$ t.u. must be added. ■

Remark 4.4: Since $d < n$, some will want me to simplify the bound $O(\log(d) + \log(n))$ to $O(\log(n))$, and they will be right. However, I keep my notation, because as a good physicist from the land of Pagnol, I consider that *the big O of* $\log(d)$ *matters more than the big O of* $\log(n)$ (cf Section 4.3.3.1).

Other acyclic preferences

For other types of acyclic preferences, the key to estimating the average convergence time is to estimate the number of boiling peers. For real preferences such as those from Meridian, this is a difficult exercise. Similarly for geometric preferences, where strong correlations exist between the preferences of nearby peers. Note that for this latter case, ignoring these correlations reduces to the case of random acyclic preferences, and this works fairly well in practice (this technique will moreover be extensively used in the following section).

In *self-stabilization in preference-based systems* [55], I propose to look at the distribution of peer "value" to determine whether the convergence of a system is closer to that of global preferences or to that of acyclic preferences. If some peers are well ranked by many of their neighbors (they are thus "good" peers), the convergence should resemble that of global preferences ($O(d)$ or $O(d + \log(n))$). On the other hand, if no good peer stands out (peers all have roughly the same value), the convergence should be of the random acyclic type ($O(\log(d))$ or $O(\log(d) + \log(n))$).

I suggest using this rule of thumb (every acyclic preference network has behavior between that of random acyclic preferences and that of global preferences, the cursor being positioned by the distribution of values) for all types of acyclic preferences, for instance for linear combinations of preferences. And also for arbitrary acceptability graphs and quotas. Unfortunately, I have no theoretical proof to offer. Fortunately, for everything I cannot prove, there are simulations.

4.3.3 Simulations

Beyond purely theoretical results, I propose simulations to complete the understanding of the convergence of acyclic preferences. This is an approach I am particularly fond of when a system that depends on a few parameters cannot be fully analyzed: varying the parameters one by one in order to build an intuition about the influence of each of these parameters. All results presented here come from the article *self-stabilization in preference-based systems* [55], to which I invite the reader concerned with technical details to refer.

Best partner **initiative**

To begin, let us look at convergence under the *best partner* initiative, which was extensively studied at the beginning of this section. Quotas greater than 1 will be used to see how the results seen for $b = 1$ apply for $b \geq 1$.

Size of the acceptable neighborhood

In order to understand the influence of the number of neighbors, we fix n and vary $d = p(n-1)$ (G is still assumed to be Erdős-Rényi). Two main behaviors are observed:

- For global preferences, the convergence time grows linearly; the round-robin sequence is faster than the Poissonian sequence except when d is close to n;
- for the other preferences studied, the convergence time grows in a seemingly logarithmic manner; the round-robin sequence is always faster than the Poissonian sequence; the two fastest preferences (random acyclic and geometric) are virtually indistinguishable.

System size

Let us now consider systems where n varies, with d constant. The results are as follows:

- The convergence time increases with n for Poissonian sequences. This is the *endgame* effect in $O(\log(n))$;
- for round-robin sequences, the convergence time increases noticeably less. There is even a slight decrease for global preferences when n is close to d;
- except for the case above (global preferences, n close to d), Poissonian sequences are longer than round-robin sequences;
- as in the previous experiment, acyclic and geometric preferences converge the fastest, followed by Meridian preferences and finally, relatively far behind, global preferences.

The main lesson to take away from these simulations is that for fixed d, n has relatively little effect on the convergence time. “The big O of d is more important than the big O of n”.

Collaboration quotas

As for the other parameters, the role of quotas in convergence time depends on the type of preferences:

- For global preferences, the convergence time first grows quickly then slowly (the first quotas are the most expensive); Poissonian sequences are slightly faster;

- for the other preferences, a linear or quasi-linear growth is observed, with round-robin sequences being slightly faster.

Summary

In summary, global preferences are distinguished by a convergence time that is linear with respect to the average degree, making d the main parameter to consider for convergence. The quotas b have a weaker influence, especially once the first values are past (the more connections there are, the less time a new connection takes).

For the other preferences, where there are not really any "good" peers, it is almost the opposite: convergence is proportional to the quotas (each connection costs the same amount of time as the previous one), but only logarithmic with respect to the neighborhood size.

Random and hybrid initiatives

Most of the work I have done so far concerns the *best partner* initiative, as it is a deterministic initiative, which simplifies its study. To conclude this discussion on convergence, I would like to briefly address the *random* initiative, which I believe will be a very interesting subject to study in future work.

A *random* initiative is much less costly in control messages than a *best partner* initiative, because it is unnecessary to know the ranks and partners of one's neighbors to implement it. In return, it increases the convergence time: while a boiling peer necessarily stabilizes an edge by taking a *best partner* initiative, this is no longer the case with a random initiative.

However, it is not only the time required for complete convergence that matters. The time needed to reach *good enough* configurations (for some measure) is at least as important.

For example, it is possible to measure peer satisfaction in a given configuration (cf [55] for a detailed description) and to observe how this satisfaction evolves during the convergence process. For non-global preferences (geometric, random or Meridian) and the same initial conditions, the *best partner* initiative produces at each instant a better satisfaction than the random initiative. On the other hand, the situation is much less clear-cut if the preferences are global: on the one hand, the random initiative needs only a few t.u. to create a high satisfaction, but then struggles to reach the stable satisfaction; on the other hand, the *best partner* initiative has a much slower initial growth and only surpasses the satisfaction of the random initiative late, but then converges quickly.

The interest of this example is to show that in addition to being less costly, there are cases (global preferences in this instance) where the random initiative can prove more efficient: having a very strong initial convergence, even if the final convergence is slow, can be very advantageous for a system subject to high churn (and where it is therefore illusory to try to achieve complete convergence). I take the opportunity to note that the BitTorrent protocol, which can very roughly be likened to a system with global preferences subject to churn, uses the random initiative, better known here as *optimistic unchoking*.

I propose the following interpretation (also drawn from [55]) to explain this difference in convergence for global preferences: with the *best partner* initiative, everyone tries the “good” peers. In particular, the bad peers will continuously try the best non-stabilized peers. The latter stabilize very quickly (they are hot), which breaks their bad connections. The result is a kind of convergence front (or saturation front) that moves temporally from the best to the worst peers, with peers being stabilized after the front passes, but having few or no connections before. In particular, a bad peer cannot retain its partners before the complete convergence of the system. Conversely, the random initiative does not create a saturation front, but rather a sort of uniform convergence of satisfaction: bad peers can choose bad partners, which means that their intermediate connections last longer.

The advantages of *random* and *best partner* initiatives can be combined in hybrid initiatives. I propose for example an initiative where each peer operates in random initiative mode below a certain number of partners, and switches to *best partner* beyond that. The good peers converge almost as fast as with *best partner*, following a saturation front, but the peers ahead of the front still have good satisfaction due to the random component that kicks in whenever there are not enough partners. In the end, the hybrid initiative performs well regardless of the acyclic preferences (not just for global preferences therefore), both for initial and complete convergence, making it an interesting choice, particularly for acyclic systems where the exact nature of the preferences is unknown or variable.

4.4 Stable configurations

Before concluding this chapter on networks with acyclic preferences, I propose to answer, at least partially and for Erdős-Rényi acceptability graphs, the second major question of acyclic preference systems: *what are the properties of the stable configuration?*

For $b = 1$, it is possible to study the distribution of partners through a (simplified) *mean field* approach. For global preferences, one can prove the existence of a fluid limit with an explicit solution, which shows in particular that the distribution of the value of a peer’s partner is centered around the value of that peer, with exponential decay: this is the *stratification* effect [37]. For random acyclic or geometric preferences, the solution of the fluid limit decays as a power law. Although I am unable to prove the *mean field* approach, I propose to validate it by comparing it with the exact solution (for global preferences) or using simulations.

Finally, I propose to extend the results to $b \geq 1$. The fluid equations then no longer seem to admit explicit solutions, but the same asymptotic behavior as for $b = 1$

is observed (exponential for global preferences, power law otherwise). An unexpected consequence is that for geometric preferences, the stable configuration is a small-world if the quotas are sufficient.

Most of the results presented here are gathered in the article *The Stable Configuration in Acyclic Preference-Based Systems* [57], of which an extended version is available as a research report [58].

4.4.1 Specific notation

Since I propose to study the stable configuration in detail in this section, some additional notation is necessary, particularly to describe the distribution of partners. If i and j are acceptable, $r_i(j)$ denotes the acceptable rank of j according to i (1 being the best rank). r_i is the acceptable ranking of i. If i has more than k (acceptable) neighbors, $r_i^{-1}(k)$ is the kth best acceptable neighbor of i. Similarly, for any $j \neq i$, $R_i(j)$ denotes the rank of j in the complete graph, without taking into account the notion of acceptability[8]. R_i is the *complete* ranking of i. For $K < n$, $R_i^{-1}(K)$ is the Kth best neighbor, acceptable or not, of i.

In all that follows, D denotes the distribution of the stable partner or partners. To lighten the notation, I propose to use a loose notation, where the meaning of D is specified using subscripts and superscripts whenever necessary. Thus, $D_{R_i}(K)$ indicates the probability that i has its peer of complete rank K as a partner; $D_{n,d}(i,j)$ is the probability that i and j are partners if there are n peers with Erdős-Rényi acceptability of average degree $d = p(n-1)$; if $c \leq b(i)$, $D_{r_i,c}(k)$ is the probability that the cth best stable partner of i has acceptable rank k…

The complementary cumulative distribution function (CCDF) of D is denoted S, and the normalized versions of D and S are respectively denoted $\mathcal{D}$ and $\mathcal{S}$.

4.4.2 Acyclic equations

For the case of simple matching ($b = 1$), I propose a generic method to describe the complete rank of $C(i)$, if it exists, in the stable configuration C. The generalization to b-matching will be done in Section 4.4.5.

Exact equation

Let $D_{R_i}(K)$ be the probability that $R_i(C(i)) = K$, i.e. that the partner of i, if it exists, has complete rank K. The CCDF of D is defined by $S_{R_i}(K) := 1 - \sum_{L=1}^{K-1} D_{R_i}(K)$, i.e. the probability that i has a partner of complete rank greater than or equal to K ($R_i(C(i)) \geq K$) or has no partner at all (shorthand notation for the

[8] I assume the existence of a natural extension of the value matrix to non-acceptable pairs. This extension is immediate for global preferences (the intrinsic value) and geometric preferences (the distance). For random acyclic preferences, it is obtained by assigning random “phantom” values to non-acceptable edges.

disjunction of both events: $R_i(C(i)) \nless K$). Inspired by the approach used in [37], I propose an exact equation describing D_{R_i} before giving a simplified version derived from a *mean field* approach.

To express $D_{R_i}(K)$, one can observe that for i to be stable with its peer of complete rank K, denoted $j := R_i^{-1}(K)$, the following three conditions must be (and suffice to be) satisfied:

- $\{i, j\}$ must be an acceptable edge, which occurs with probability p if G is an Erdős-Rényi graph $\mathcal{G}(n, p)$;
- i must not have a better stable partner than j ($R_i(C(i)) \nless K$);
- j must not have a better stable partner than i ($R_j(C(j)) \nless R_j(i)$).

This leads to the following exact equation:

$$D_{R_i}(K) = pS_{R_i}(K)\,\Pr\big(R_j(C(j)) \nless R_j(i) \mid R_i(C(i)) \nless K\big) \tag{4.1}$$

Approximate equation

The main difficulty of Equation (4.1) is the conditional probability, which is delicate to handle because of the correlations that may exist between $R_j(C(j)) \nless R_j(i)$ and $R_i(C(i)) \nless K$. The solution is to consider these correlations as negligible:

> **Approximation 4.1**: The events "i is not with someone better than j" and "j is not with someone better than i" are independent.

This approximation, which I sometimes somewhat loosely call "mean field hypothesis", is reasonable for p small enough[9]. Equation (4.1) can then be simplified to

$$D_{R_i}(K) = pS_{R_i}(K)S_{R_j}\big(R_j(i)\big) \tag{4.2}$$

To go further, one must take into account the type of preferences.

4.4.3 Global preferences

Since global preferences are characterized by a total order on the peers, one can dispense with the value matrix m by labeling the peers from 1 to n, 1 being the best. We thus directly consider the probability $D(i, j)$ that i and j are partners. Noting that the complete rank of j for i is j if $j < i$ and $j - 1$ if $j > i$ (a peer does not rank itself), we obtain the relation between D and D_R:

$$D(i, j) = \begin{cases} D_{R_i}(j) & \text{si } j < i, \\ 0 & \text{si } j = i \quad \text{(non-réflexivité)}, \\ D_{R_i}(j-1) & \text{si } j > i. \end{cases} \tag{4.3}$$

[9] A few simple examples seem to indicate an error of order p^3, confirmed by the simulations [37].

Setting $S(i,j) := 1 - \sum_{k=1}^{j-1} D(i,k)$, we obtain the version of Equation (4.2) for global preferences:

$$D(i,j) = \begin{cases} 0 & \text{si } i = j, \\ pS(i,j)S(j,i) & \text{sinon.} \end{cases} \tag{4.4}$$

This equation[10], which is solved numerically by a double iteration, gives a very good approximation of the empirical distribution [37].

Normalization

An explicit fluid limit toward which the discrete distributions converge is a definite asset for describing the stable configuration. It indeed allows giving a complete, immediate and global description of the distribution for all n and p, which is not the case of Equation (4.4).

In order to obtain this limit, one must be able to compare distributions for arbitrary values of n. It is therefore necessary to normalize D. A fairly simple way to do this is to represent each peer i by a normalized rank α, with $0 \leq \alpha < 1$. More precisely, we associate to each i the real number $\alpha(i) = \frac{i-1}{n}$, and conversely to each positive real α the integer $i(\alpha) = \lfloor n\alpha \rfloor + 1$. The normalized version of D, denoted $\mathcal{D}$, is then defined by

$$\mathcal{D}_n(\alpha,\beta) = nD(\lfloor n\alpha \rfloor + 1, \lfloor n\beta \rfloor + 1) \tag{4.5}$$

$\mathcal{D}_n$ is a step function (of two variables), which takes the values $(nD(i,j))$. The factor n allows expressing $D(i,j)$ simply as an integral of $\mathcal{D}$:

$$D(i,j) = \int_{\frac{j-1}{n}}^{\frac{j}{n}} \mathcal{D}_n\left(\frac{i-1}{n}, x\right) \mathrm{d}x = \int_{\frac{i-1}{n}}^{\frac{i}{n}} \mathcal{D}_n\left(x, \frac{j-1}{n}\right) \mathrm{d}x \tag{4.6}$$

The normalized complementary cumulative distribution is defined by

$$\mathcal{S}_n(\alpha,\beta) = 1 - \int_0^\beta \mathcal{D}_n(\alpha,x)\,\mathrm{d}x \tag{4.7}$$

and the relation between S and $\mathcal{S}$ is

$$S(i,j) = \mathcal{S}\left(\frac{i-1}{n}, \frac{j-1}{n}\right) \tag{4.8}$$

Convergence of normalized distributions

If the average degree remains constant, I have shown the existence of a continuous limit of the distributions $\mathcal{D}$. The first step is to address the problem of an intrinsic discontinuity along the main diagonal ($\alpha \approx \beta$), due to the fact that $D(i,i) = 0$. This is a minor problem since this discontinuity is simply there to recall the non-reflexivity of

[10] For the record, Equation (4.4) was proposed by Julien Reynier based on an original idea by Fabien de Montgolfier. Chronologically it is the origin of Equation (4.2) (general equation), and not the other way around.

the matching. It is resolved by introducing a more "continuous" function $\mathcal{D}$, obtained by extending $\mathcal{D}$ on the main diagonal using Equation (4.4):

$$\tilde{\mathcal{D}}(\alpha,\beta) = \begin{cases} \mathcal{D}(\alpha,\beta) & \text{si } \lfloor n\alpha \rfloor \neq \lfloor n\beta \rfloor, \\ np(S(\lfloor n\alpha \rfloor + 1, \lfloor n\alpha \rfloor + 1))^2 & \text{sinon.} \end{cases} \tag{4.9}$$

The fluid limit, at constant degree, of the functions $\tilde{\mathcal{D}}$ is then given by the following theorem:

Theorem 4.9: Soit $d > 0$ une constante. Si $n \to \infty$, avec $p = \frac{d}{n}$, les fonctions $\tilde{\mathcal{D}}_{n,d}$ convergent uniformément vers

$$\mathcal{D}_\infty(\alpha,\beta) = \frac{de^{d|\beta-\alpha|}}{\left(1 - e^{-d\min(\alpha,\beta)} + e^{d|\beta-\alpha|}\right)^2} \tag{4.10}$$

This result shows that asymptotically, the distribution of partners depends only on the average degree of G (assumed Erdős-Rényi). This allows quantitatively describing the *stratification* effect [37]: for fixed α, the distribution of the stable partner of α decays exponentially in $|\beta - \alpha|$, with intensity d ($\mathcal{D}_\infty(\alpha,\beta) \approx de^{-d|\beta-\alpha|}$ for $d|\beta-\alpha|$ large enough). In other words, a peer of normalized rank α tends to have as partner a peer of the same rank, within plus or minus $\frac{1}{d}$.

Proof: The proof of Theorem 4.9 consists of four steps [58]:

- show that the functions $\tilde{\mathcal{D}}_{n,d}$ are uniformly Cauchy on $[0,1]^2$;
- use Cauchy convergence to show that $\mathcal{S}_N$ and $\tilde{\mathcal{D}}_N$ admit limits $\mathcal{S}_\infty$ and $\mathcal{D}_\infty$;
- give a PDE satisfied by $\mathcal{S}_\infty$;
- solve the PDE[11], and deduce $\mathcal{D}_\infty$ from the solution.

■

The existence of this fluid limit had been proposed as a conjecture as early as [37], and proven for the case $\alpha = 0$, but it was only later that the complete proof and expression were found [58].

Theorem 4.9 yields three immediate corollaries that complete the understanding of the stable configuration.

Corollary 4.1: Considering $\mathcal{S}_\infty(\alpha, 1)$, the probability that a peer of normalized rank α is unmatched in the stable configuration is $\frac{1}{1+e^{-d\alpha}(e^{-d}-1)}$.

[11] For the reader eager for "exotic" equations, this involves solving $\partial_\beta \mathcal{S}_\infty(\alpha,\beta) = -d\mathcal{S}_\infty(\alpha,\beta)\mathcal{S}_\infty(\beta,\alpha)$, with the initial condition $\mathcal{S}_\infty(\alpha,0) = 1$. Although seemingly innocent, solving this non-local equation gave me quite a hard time, even though I already knew its solution by other means. I take this opportunity to thank François Baccelli who pointed me toward the approach of de-non-localization.

Corollary 4.2: For $i \neq j$ (discrete case), a good approximation of $D(i,j)$ is

$$D(i,j) \approx \frac{pe^{p|j-i|}}{\left(1 - e^{-p\min(i,j)} + e^{p|j-i|}\right)^2} \tag{4.11}$$

Corollary 4.3: Let a sequence of normalized distributions $\mathcal{D}_{n,d}$ with unbounded increasing degree ($d \overset{n\to\infty}{\to} \infty$). We have

$$\forall \alpha \in [0,1[, \quad \mathcal{D}_{n,d}(\alpha,.) \overset{(n\to\infty)^*}{\to} \delta_\alpha \tag{4.12}$$

This last corollary generalizes a theorem proposed in [37], which showed the existence of a weak Dirac limit in the case of a sequence of distributions with $p = \frac{d}{n-1}$ constant. It means that as soon as the degree tends to infinity (for instance as $O(\log(n))$), then asymptotically, a peer has as partner a peer of the same normalized rank (no deviation), and the probability of being unmatched becomes zero.

Validation

To validate the previous results, one must compare simulation results with Equation (4.4) (recurrence under the independence approximation), then with Equation (4.11). This is what was done in [37, 58].

As indicated previously, we observe for Equation (4.4) a very good precision, of order p^3.

For the fluid limit, apart from the continuous extension on the main diagonal, the precision is also very good, except for small values of n and high values of d. These observations are consistent with the complete proof of Theorem 4.9 [58], which shows convergence in $O\left(\frac{d^2}{n} \cdot e^{8d}\right)$. They would even rather support convergence in $O\left(\frac{d^2}{n}\right)$, which is in my opinion the true bound, although this remains to be proven[12].

Exact resolution

In the situation I have just proposed ($b = 1$, global preferences, acceptability $\mathcal{G}(n,p)$), it is in fact unnecessary to resort to the independence approximation, since there exists a recurrence that yields the exact distribution [58].

I was able to obtain this recurrence by conditioning the fact that i and j are partners according to the rank of the partner of peer 1 (less than i, between i and j, greater than j, or no partner).

[12] The factor e^{8d} comes from the use of Grönwall's lemma [12], which amplifies an incompressible quantization error of $O\left(\frac{d^2}{n}\right)$. But in practice, Equation (4.4), which serves as the basis for constructing the distributions, is self-stabilizing (an error in one direction at a given moment is compensated in the other direction at the next iteration), which Grönwall's lemma does not allow to take into account.

Just as for the approximate formula, the exact formula admits a fluid limit, which obeys a certain PDE. Although the two PDEs (exact and approximate) are extremely different[13], they give the same solution, namely Equation (4.10).

The fact that in this particular case, the recurrence given by the independence approximation has exactly the same fluid limit as the exact recurrence is a strong argument (albeit not a rigorous one) for justifying the use of this approximation in other cases. For a major drawback of the exact recurrence is that the "trick" used does not apply to other preferences, nor for $b > 1$. However elegant this recurrence may be, the independence hypothesis therefore remains indispensable if one wants to generalize the results.

4.4.4 Acyclic and geometric preferences

I now propose to turn to (random) acyclic and geometric preferences. For these preferences, one can note that the peers are undifferentiated. Over the set of possible realizations, one can therefore consider that all peers follow the same partner distribution: $D_{R_i}(K)$ is independent of i, and can therefore be denoted $D_R(K)$.

As for global preferences, the goal is to find the distribution of the complete rank by simplifying Equation (4.1) before solving it. I also briefly give the distribution of distances and elements for solving the distribution of the relative rank.

Distribution of the complete rank
For geometric or random preferences, the Approximation 4.1 is not sufficient, which is why I propose an additional approximation:

> **Approximation 4.2**: The complete rank is symmetric: $R_i(j) = R_j(i)$.

I thus assume that $R_i(j)$ is not too bad an approximation of $R_j(i)$ for the preferences considered. This gives a very simple equation for $D_R(K)$:

$$D_R(K) = pS_R^2(K), \quad \text{avec } S_R(K) = 1 - \sum_{L=1}^{K-1} D_R(L) \tag{4.13}$$

S_R can thus be obtained by a simple recurrence:

$$S_R(K) = \begin{cases} 1 & \text{si } K = 1, \\ S_R(K-1) - pS_R^2(K-1) & \text{sinon.} \end{cases} \tag{4.14}$$

And D_R is directly given by $D_R(K) = S_R(K) - S_R(K+1)$.

Fluid limit.

[13]The exact PDE is relatively classical and is solved by the method of *characteristics*, whereas the approximate PDE, described in an earlier note, had to be solved in a less conventional manner.

As for global preferences, D_R can be normalized. For $0 \leq \alpha < 1$, we set $\mathscr{d}_R(\alpha) := (n-1)D_R(\lfloor (n-1)\alpha \rfloor + 1)$. The normalization factor is now $n-1$ since this is the maximum value of K (n was that of i and j in Section 4.4.3). D_R is expressed as an integral of $\mathscr{d}_R$:

$$D_R(K) = \int_{\frac{K-1}{n-1}}^{\frac{K}{n-1}} \mathscr{d}_R(x)\,\mathrm{d}x \tag{4.15}$$

$\mathscr{s}_R$ is then naturally defined by:

$$\mathscr{s}_R(\alpha) = 1 - \int_0^{\alpha} \mathscr{d}_R(x)\,\mathrm{d}x \tag{4.16}$$

Theorem 4.10: Soit $d > 0$ une constante. Si $n \to \infty$, avec $p = \frac{d}{n}$, les fonctions $\mathcal{S}_R$ convergent uniformément vers

$$\mathscr{s}_\infty(\alpha) = \frac{1}{d\alpha + 1} \tag{4.17}$$

En particulier, la probabilité qu'un pair soit célibataire dans la configuration stable est asymptotiquement $\mathscr{s}_R(1) = \frac{1}{d+1}$, et une bonne approximation de $S_R(K)$ est

$$S_R(K) = \frac{1}{p(K-1)+1} \tag{4.18}$$

Proof: The proof is the same as for Theorem 4.9, but easier. One must first show that the $\mathcal{D}_R$ are uniformly Cauchy (which is easier here since there is only one variable and no need to introduce an extension on the main diagonal). We then have uniform convergence toward a continuous function $\mathscr{s}_\infty$. We then deduce from Equation (4.14) an ordinary differential equation satisfied by $\mathscr{s}_\infty$:

$$-\dot{\mathscr{s}}_\infty(\alpha) = d\mathscr{s}_\infty^2(\alpha) \tag{4.19}$$

with the initial condition $\mathscr{s}_\infty(0) = 1$. The solution of Equation (4.19) is Equation (4.17), which completes the proof. ■

Validation.

Given the approximations made, it is necessary to verify on a few examples the precision of Equation (4.18). I have therefore tested its validity for a few values of n and p [58].

For $p = 1$, one cannot really say that the independence hypothesis is satisfied. Simulations show that the distributions are affected by the particular type of preferences, and the fluid limit is not very precise, particularly at the boundaries (K close to 1 or n). One can among other things note that the probability of being unmatched

$\frac{1}{n}$ given by Equation (4.18) is clearly overestimated if n is even (since it is then actually zero), but exact for odd n. The fluid limit nevertheless manages to give the $\frac{1}{K}$ behavior common to all preferences considered. From this point of view, it is better than the recursive (4.14) (from which it derives), which gives $S_R(K) = \delta_K^1$ for $p = 1$.

As p decreases, the empirical curves and the limit given by Theorem 4.10 converge very quickly. By $p = \frac{1}{10}$, they are almost indistinguishable, which validates the use of the fluid limit as an approximation of the complete rank distribution.

Distribution of distances

It may be interesting to consider distributions other than that of the complete rank. For example, for geometric preferences, the distribution of distances between stable partners can be interesting if the performance of a matching is related to distances. Now, this distribution can be deduced from that of the complete rank:

Theorem 4.11: Soit $S_X(x)$ la probabilité qu'un pair n'ait pas de partenaire stable à distance inférieure à x dans le tore unitaire de dimension t. Soit B_t le volume d'une boule de rayon x dans ce tore. Sous la limite fluide, on a

$$S_X(x) = \frac{1}{dB_t(x) + 1} \tag{4.20}$$

Proof: A ball of radius x, centered on any peer, contains approximately $nB_t(x)$ peers since it occupies a proportion $B_t(x)$ of the torus (which is assumed to be unit). The most distant peer inside this ball should therefore have a complete rank of approximately $nB_t(x)$ for the central peer, while being at a distance of approximately x. We thus obtain the relation $S_X(x) = S_R(nB_t(x))$, and it only remains to use Equation (4.18) to conclude. ■

The function $B_t(x)$ depends on the dimension t and the norm used. For the infinity norm, we simply have $B_t(x) = \min((2x)^t, 1)$. The expression for other norms can be somewhat more complicated due to possible wrap-around effects at the boundaries. Note that if I had chosen $\mathbb{R}^t$ (with homogeneous density) instead of the unit torus, $B_t(x)$ would simply have been the volume of a ball of radius x, without boundary effects.

As usual, the precision of Equation (4.20) has been verified empirically, and the result is that it has virtually the same region of validity as Equation (4.18) [58].

Distribution of the relative rank

For random acyclic and geometric preferences, I have also studied the distribution of the relative rank, given by the function $D_r(k)$ and its complementary cumulative distribution $S_r(k) := 1 - \sum_{l=1}^{k-1} D_r(k)$.

I therefore tried to adapt the method used for the complete rank. The conditions to be fulfilled for the stable partner of i to be its kth acceptable neighbor $j = r_i^{-1}(k)$ are the following:

- i must have at least k neighbors (if the kth neighbor exists, it is acceptable by definition),
- i must not be with a peer better than j,
- j must not be with a peer better than i.

By adapting the independence and symmetry approximations to the relative rank, we then obtain the following recurrence formula [58]:

$$D_r(k) = S_r(k)\frac{1 - I_{1-p}(n-k+1, k)}{k+1} \tag{4.21}$$

where I_x is the regularized incomplete beta function.

Unfortunately, Equation (4.21) is much less precise than Equation (4.18) is for the complete rank, particularly for $D_r(1)$ [58]. The reason is that correlations are much more present when it is the acceptable rank that is considered (particularly if one considers the close neighborhood).

As an exercise, I therefore asked myself whether it was possible to refine the estimate of the relative rank, and I partially succeeded: for $D_r(1)$, it is possible to have a better estimate in the fluid limit, which is obtained by conditioning $D_r(1)$ on the normalized complete rank of the first acceptable partner [58]. We then obtain

$$D_r(1) = e \cdot E_1(1) \approx 0.596 \tag{4.22}$$

where E_1 is the exponential integral.

This value is very good in almost all cases for random acyclic preferences. For geometric preferences, it is however necessary to be in good conditions, i.e. small p and large n [58].

4.4.5 Generalization to b-matching

I now propose to extend the previous results to the b-matching problem. b is still assumed to be constant across peers for simplicity. I only propose here results on the complete rank, although it is a priori possible to reuse the techniques seen previously for the relative rank or distances.

Equations under the independence hypothesis

A peer can now have up to b partners. We therefore denote, for $1 \leq c \leq b$, D_c the distribution of the complete rank of the cth best stable partner, and S_c the corresponding CCDF. Just as for the case $b = 1$, it is possible to give the necessary and sufficient conditions for $j = R_i^{-1}(K)$ to be the cth best stable partner of i:

- the pair $\{i, j\}$ must be acceptable,

- the $(c-1)$th best partner of i (if $c > 1$) must be strictly better than j, while the cth (if it exists) must not be,
- the bth partner of j, if it exists, must not be better than i.

By extending the independence hypothesis Approximation 4.1, we thus obtain a generic formula for b-matching:

$$D_{R_i,c}(K) = \begin{cases} pS_{R_i,1}(K)S_{R_j,b}\big(R_j(i)\big) & \text{si } c = 1\text{\textbackslash, sinon} \\ p\big(S_{R_i,c}(K) - S_{R_i,c-1}(K)\big)S_{R_j,b}\big(R_j(i)\big). \end{cases} \tag{4.23}$$

To be usable, this formula must be adapted to the type of preferences considered.

For global preferences, if $D_c(i,j)$ denotes the probability that the cth stable partner of i is j[14], we obtain the following recursive system, which is solved by a triple iteration over i, j and c [37]:

$$D_c(i,j) = \begin{cases} 0 & \text{si } i = j\text{\textbackslash, sinon} \\ pS_1(i,j)S_b(j,i) & \text{si } c = 1, \\ p(S_c(i,j) - S_{c-1}(i,j))S_b(j,i) & \text{si } c > 1. \end{cases} \tag{4.24}$$

Similarly, for random acyclic and geometric preferences, we obtain the following system from Approximation 4.2 (symmetry of the complete rank) [58]:

$$D_{R,c}(K) = \begin{cases} pS_{R,1}(K)S_{R,b}(K) & \text{si } c = 1, \\ p\big(S_{R,c}(K) - S_{R,c-1}(K)\big)S_{R,b}(K) & \text{si } c > 1. \end{cases} \tag{4.25}$$

Using $S_{R,c}(1) = 1$ and $D_{R,c}(K) = S_{R,c}(K) - S_{R,c}(K+1)$, system Equation (4.25) is easily solved by a double iteration over K and c.

Simulations show that both systems agree very well with the empirical distributions as long as n and p satisfy the usual conditions [58]. The behavior is qualitatively very similar to that of simple matching: exponential decay for global preferences (but because of the multiplicity of partners, shifts appear between the density peaks of the D_c and the value of the original peer), heavy tail distribution for geometric and random acyclic preferences.

For $b > 1$, fluid limits also exist. Unfortunately, unlike simple matching, I have not managed to find explicit solutions to describe them (and as time passes, I increasingly doubt that such solutions exist). It is nevertheless possible to give the PDEs satisfied by these limits.

For global preferences, the fluid limits satisfy

$$\partial_y \mathcal{S}_c(\alpha,\beta) = \begin{cases} -d\mathcal{S}_1(\alpha,\beta)\mathcal{S}_b(\beta,\alpha) & \text{si } c = 1\text{\textbackslash, sinon} \\ -d(\mathcal{S}_c(\alpha,\beta) - \mathcal{S}_{c-1}(\alpha,\beta))\mathcal{S}_b(\beta,\alpha). \end{cases} \tag{4.26}$$

with the initial conditions $\mathcal{S}_c(\alpha,0) = 1$.

[14] I take the opportunity to point out that D_c is no longer symmetric, whereas it was for $b = 1$.

Similarly, for acyclic and geometric preferences, the limits $\mathscr{s}_{R,c}$ satisfy

$$\dot{\mathscr{s}}_{R,c} = \begin{cases} -d\mathscr{s}_{R,1}\mathscr{s}_{R,b} & \text{si } c = 1, \\ -d\big(\mathscr{s}_{R,c} - \mathscr{s}_{R,c-1}\big)\mathscr{s}_{R,b} & \text{si } c > 1. \end{cases} \tag{4.27}$$

with the initial conditions $\mathscr{s}_{R,c}(0) = 1$.

Even if these equations cannot be solved completely, the fluid limit still has several advantages.

First, the very existence of a limit allows computing it numerically with precision and using the result for multiple values of n and d. Take for example the case of global preferences (the same reasoning can be applied to acyclic and geometric preferences). We assume b is fixed. Let $d_{\max}$ be the maximum average degree of the distributions we want to evaluate, and N a fixed sampling size of the fluid limit (the larger N, the more precise the evaluation). We can then set $p = \frac{d_{\max}}{N-1}$ and compute the $D_{N,p,c}$ (once and for all) using Equation (4.24). For a degree $d \leq d_{\max}$, we set $N' = \frac{d}{p}$ (N' is not necessarily an integer). Inspired by the normalization Equation (4.5), we obtain the following approximation for $|\alpha - \beta| \geq \frac{1}{N'}$:

$$\mathscr{D}_{d,c}(\alpha, \beta) \approx N' D_{N,p,c}(\lfloor N'\alpha \rfloor + 1, \lfloor N'\beta \rfloor + 1) \tag{4.28}$$

We can then in turn use this estimate of the fluid limit for discrete distributions. Thus, for any integer n and for any $d \leq d_{\max}$, we have, for $|i - j| \geq \frac{n}{N'}$ (which amounts to $i \neq j$ if $n \leq N'$)

$$D_{n,d,c}(i,j) \approx \left(\frac{N'}{n}\right) D_{N,p,c}\left(\left\lfloor \left(\frac{N'}{n}\right)(i-1) \right\rfloor + 1, \left\lfloor \left(\frac{N'}{n}\right)(j-1) \right\rfloor + 1\right) \tag{4.29}$$

Another advantage of fluid limits is that the PDEs they satisfy give us information about their behavior. One can for example use them to show that $\mathscr{s}_{R,1} \leq \mathscr{s}_R \leq \mathscr{s}_{R,b}$ (and the equivalent for global preferences) and thus understand why the behavior in b-matching remains similar to that of simple matching.

4.4.6 Some applications

Having put considerable effort into the mere study of distributions, I must admit I have not yet devoted much time to advanced properties of stable configurations, i.e. those likely to help understand existing systems and develop new ones. Here are nevertheless two: the stratification of global preferences and the "small-worldification" of geometric preferences.

Stratification

As we have just seen, with global preferences, the stable partners of a given peer i have, on average, the same rank as i. This is stratification, which guarantees a certain fairness in the stable configuration [37]: in terms of rank, what a peer gives should be roughly equal to what it receives. One must also remember that the $\mathscr{D}_c(\alpha, .)$ have exponential decay with deviation of $\frac{1}{d}$, d being the average degree of the acceptability graph. One then realizes that the following trade-off must be resolved:

- if d is too small the deviation is high. In particular, if the entries of the value matrix (for example the bandwidths of a *BitTorrent*-like system) follow a non-uniform distribution, there can be a very large difference between the expected gain and what one gives. This problem was highlighted in [37] to explain potential flaws in the Tit-for-Tat technique employed by BitTorrent;
- a large d on the other hand strengthens fairness. But in practice, increasing the size of the acceptability graph has a cost for the peers: convergence time, memory space, graph maintenance… A large d also shortens the deviation (down to the Dirac in the fluid limit), and therefore increases the diameter of the stable configuration, which can be problematic if one wants to propagate information through stable edges.

For the quota b, which represents the maximum degree in the stable configuration, a similar trade-off exists: a large b can improve fairness and decrease the diameter, but will be costly in resources.

This suggests that for most systems with global preferences (i.e. based on the sharing of bandwidth, storage capacity, computing power, uptime…), there should exist a pair (d, b) (or more generally a coupling between an acceptability graph and a vector of quotas) optimal for the stable configuration, whose exact value would depend on the importance given to parameters such as diameter, fairness, convergence time, or maintenance cost.

Small-worldification

A small-world is a sparse graph (average degree in $O(\log(n))$ or even $O(1)$) with an average shortest path length (ASPL) in $O(\log(n))$ and a high clustering coefficient (there are far more short cycles than for a random graph of the same size). For example, Kleinberg showed some years ago that an n-dimensional grid could be transformed into a small-world by adding long-range edges following a distribution in $\Omega\left(\frac{1}{x^n}\right)$ [49].

Let us now look at what happens for a stable configuration, with b in $O(\log(n))$ so that the configuration is “sparse”. If it is global preferences, the clustering is indeed there as a consequence of stratification, but the diameter tends to grow linearly [36]. Similarly, for random acyclic preferences, the small diameter is verified but there is no high clustering (the stable configuration behaves like an incomplete random b-graph). On the other hand, for geometric preferences, the heavy tail distribution allows having both properties: on the one hand, most stable partners have a small complete rank; they are therefore “geographically” close which gives clustering; on the other hand, there exist long-range links that make the diameter small. Geometric preferences are therefore conducive to generating small-world configurations, and this is indeed what occurs [36]. This opens a number of perspectives on the use of the stable configuration, starting with an easy-to-use small-world generator.

What is surprising about this small-worldification with geometric preferences is that it is solely created by the way peers rank each other: actual distances are used to construct these rankings, but the values themselves play no direct role in the preference system in general and in its stable configuration in particular. This leads one to think that topological characteristics of a system can therefore be contained in a set of preferences.

As an example of characteristics, I computed numerically the small-world parameters of preferences on tori for a few dimensions, the result being that the diameter and clustering tend to decrease when the dimension increases [58]. I then looked at what the Meridian latencies gave, and the parameters obtained turned out to be close to those of the 3-dimensional torus. I particularly like this unexpected result, because it seems to suggest that there exists a dimension of the Internet *in the sense of preferences*, which is about 3. I can thus add my brick to the wall of efforts made everywhere to estimate a dimension of the Internet (see for example [11]).

4.5 Conclusion

After a brief history of stable marriages, I have presented the foundations necessary for working with acyclic preferences: the specific (sometimes colorful) vocabulary, the formalism, the grand theorem of *existence, uniqueness, convergence*, and a description of the main classes of acyclic preferences, such as global or geometric preferences.

I then detailed the self-stabilization property, and more precisely the time required for complete convergence. I presented techniques that allow bounding or estimating this time for certain specific parameters, and I proposed the use of simulations to highlight the role played by each parameter. One should remember that in the worst case, with asynchronous peers, convergence can be very slow, whereas in practice, it is fairly fast (global preferences), or even very fast (most other preferences).

Finally, I described the distribution of stable partners for a few classes of preferences, by proposing an analysis of this distribution under independence assumptions, then validating and extending these results numerically. For global preferences, I thus highlighted the *stratification* effect. In figurative terms, in a world where there would be only a single criterion of beauty, the beautiful end up with the beautiful, and the ugly have little choice but to be with the ugly[15].

For geometric preferences, it is a small-world effect that occurs, due to the fact that the partner distribution is heavy-tailed, which ensures both locality and long-range links. The image is that of a world governed by homophily (*birds of a feather flock together*): most of the time, reciprocity allows collaborations between neighbors, but when a peer is rejected by its closest neighbors, it must search further and further,

[15]In the same vein, I recommend the article *Beauty and distance in the stable marriage problem*, which spices up the proposal algorithm by studying the effect of introducing a beauty criterion [18].

among potential partners who appreciate it less and less the further it must go from its own position.

In the end, I hope to have succeeded in sharing my conception of acyclic preference networks: a possible model for peer-to-peer networks, a varied collection of small mathematical problems, but also a recreational computer science. I am aware that this chapter raises more questions than it provides answers: modeling of random initiative, of agitation, study of new preferences (why not cyclic), arbitrary or even variable graphs and quotas, routing in geometric small-worlds… I will assuredly have neither the time nor the ability to answer all these problems, and consequently, if this chapter gives at least one reader the desire to take an interest in acyclic preference networks, I think it will have perfectly fulfilled its role.

For my part, I plan in the future to try to apply preference networks to practical situations, and to connect them to the work I have done on distribution problems. I believe that acyclic preferences are beginning to have the maturity needed to bring a new perspective to these problems, and I would like to verify this, as I discuss in the next and final chapter.

Chapter 5
Synthesis and perspectives

I have presented over the course of these few pages my own point of view on this new research domain that is peer-to-peer. Starting from an original definition that has the advantage of highlighting why peer-to-peer is a research domain, I have proposed a classification that allows describing, in broad strokes, the different themes that can be addressed, depending on whether one considers the problem of localization or distribution; whether one approaches it using an explicit or implicit structure; whether one places the decision at the sending end, the receiving end, or both sides; whether one takes an initial theoretical or empirical approach.

I then placed my own contributions within these themes: while one part of my work falls within the framework of a better understanding of the problems related to broadcasting, while opening the way to future and promising applications, another part is dedicated to creating a new model that brings a different perspective on the dynamics of self-structured systems in general, and of peer-to-peer in particular.

I have not discussed at all my work done in continuation of my thesis topic, PageRank [17, 26, 27, 39, 40, 60], because although dealing with a related subject, large graphs, it does not strictly fall within the scope of peer-to-peer, and it is not my main subject of study at present. Nevertheless, it is not impossible that a new stroke of chance might one day lead me to discover new themes related to this subject and completely rekindle my interest.

To return to the subject of peer-to-peer, I would like to insist one last time on the fact that the different themes are neither fixed nor compartmentalized, and that if this makes the classification of research more analog than dichotomous, I think that this is *in fine* beneficial. For example, certain graphs, such as de Bruijn graphs, were introduced in peer-to-peer in order to build DHTs [32]. Later, these same graphs escaped the localization problem and were used to propose broadcast structures [35]. Similarly, epidemic techniques, which are by nature broadcasting techniques, can have utility in localization [46]. Over the years, this porosity thus allows advancing the research front, and I like to believe that this is far from over.

For all these exchanges correspond perfectly with the philosophy I have adopted as a researcher, and my intuition tells me that by remaining attentive to the innovations and mutations occurring in all these themes, I have chances of discovering new leads that will awaken my curiosity. There are some on which I have a certain idea, and I would like to conclude this thesis by presenting these few directions on which I plan to work in the medium term, knowing that it is always possible that tomorrow, a new encounter provokes a new direction in my research.

5.1 Preference networks

As the reader will have realized, the subject of preference networks is close to my heart, and I think it should still occupy a part of my work for some time yet. The road that remains to be traveled is indeed at least as long, and I believe just as fascinating, as the one that has already been covered: finer characterization of self-stabilizing properties, modeling of dynamic graphs and quotas, study of routing properties, return to cyclic preferences… I hope, by continuing to promote it within the scientific community, to succeed in evangelizing researchers who in turn will advance the theory while developing concrete applications.

On the front of these concrete applications, I believe that it should be possible to use the model to design even more efficient epidemic broadcast algorithms. Indeed, as I have shown, global preferences allow selection by bandwidth, and recent work seems to point to bandwidth-oriented selection as promising for achieving epidemic broadcast among peers with heterogeneous capacities. Similarly, the results obtained on latency preferences allow hoping for a minimization of the network impact of broadcasting without affecting the properties of the broadcast graph. By combining these two types of preferences in the right way and applying the result to epidemic broadcasting, the final algorithm has a good chance of being very efficient.

Beyond peer-to-peer, I also hope to discover new things by looking at whether preference networks can have a connection with sensor networks, which I plan to investigate. This shift in theme will I think be beneficial for preference networks, and who knows perhaps also for sensor networks.

5.2 Prototypes

It is a fact, I am not a developer. Nevertheless, I find it extremely frustrating to have worked on concepts and algorithms without being able to verify whether they hold up in practice. If I want to be consistent with my positioning as a researcher, I therefore think that I owe it to myself to try to push the work that can be pushed as far as possible along the path to application, perhaps even to deployment. This is of course a task that I am unable to accomplish alone, and I especially hope to convince specialists in the field to carry out the development.

At present, there are two prototypes that I would appreciate having at my disposal in order to be able to test *in the field* and improve the algorithms on which I have worked so far. It is not very difficult to guess which ones: an epidemic broadcast prototype, and a video-on-demand prototype.

5.3 Future of peer-to-peer

To conclude on a more general direction, I would like to share my opinion on the future of peer-to-peer. For if peer-to-peer must evolve, research will have to adapt as well, this including my own work.

Let us return for a moment to the origins of the *golden age* of peer-to-peer, as I described them in the introduction: from a social standpoint, the explosion of peer-to-peer was triggered by the need to provide a service for which demand exists (a consequence of technical progress and the Internet bubble), but not the supply (bursting of the bubble). But today, services are once again being offered: storage spaces, file sharing and commerce or multimedia content, television and video recorder on the Internet... Thanks to superior ergonomics, these services tend to regain ground from peer-to-peer as a social phenomenon. The real stake of this battle is the localization of resources, at the user's end (peer-to-peer) or on servers (Google, to simplify). The outcome is still uncertain, with victories and defeats on both sides. What is certain is that we are in a transitional situation and that current uses, particularly peer-to-peer uses, are doomed to evolve or wither.

But *in fine*, there will always be on one side services to provide, and on the other resources, whether on a PC, a phone, a virtual video recorder, a set-top box, a proxy, or a cloud of servers. With the multiplication of platforms, on the user side or the provider side, there is therefore a question that, in my opinion, will remain relevant no matter what happens: *who retrieves what from whom?* And if I am asked my opinion on the future of peer-to-peer (and therefore incidentally on that of research on the subject), I bet on its foreseeable metempsychosis into future technologies, whatever they may be. In other words, while it is possible that peer-to-peer, in its BitTorrent-esque understanding, may end up disappearing in the coming years, peer-to-peer research, which consists of answering the question *who retrieves what from whom?* when it is not trivial, still has a bright future ahead. The name may change after absorption by a new emerging domain (Cloud Computing? Virtualization?) but I believe that the spirit, themes, and methods of the domain will survive the transition. However, this does not mean either that the domain should not anticipate its own evolution. Some of these directions are moreover already visible: appearance of new services that pose new challenges, for example *start-over* which combines slightly delayed and on-demand broadcast; growing heterogeneity of available resources, with the appearance of hybrid solutions and the multiplication of platforms; shifting of bottlenecks related to changes in the network and usage patterns.

Bibliography

1. Babelgum, http://www.babelgum.com/.
2. eDonkey, http://www.edonkey2000.com/index.html.
3. eMule, http://www.emule-project.net/.
4. Joost, http://www.joost.com/.
5. Kontiki, http://www.kontiki.com/.
6. Meridian: A Lightweight Approach to Network Positioning, http://www.cs.cornell.edu/People/egs/meridian/.
7. SopCast, http://www.sopcast.com/.
8. TVants, http://tvants.en.softonic.com/.
9. UUSee Inc., http://www.uusee.com/.
10. Abraham, D.J. et al.: The Stable Roommates Problem with Globally-Ranked Pairs. In: WINE. pp. 431–444 Springer (2007).
11. Abrahao, B., Kleinberg, R.: On the Internet delay space dimensionality. In: Proceedings of the 2008 Internet Measurement. (2008).
12. Bellman, R.: The stability of solutions of linear differential equations. Duke Math. J. 10, 643–647 (1943).
13. Benbadis, F. et al.: Playing with the Bandwidth Conservation Law. In: 8th International Conference on Peer-to-Peer Computing. pp. 140–149 (2008).
14. Bonald, T. et al.: Epidemic live streaming: optimal performance trade-offs. In: Proceedings of the International Conference on Measurement and Modeling of Computer Systems (SIGMETRICS). (2008).
15. Boufkhad, Y. et al.: Achievable Catalog Size in Peer-to-Peer Video-on-Demand Systems. In: Proceedings of the Seventh International Workshop on Peer-to-Peer Systems (IPTPS). (2008).
16. Boufkhad, Y. et al.: An Upload Bandwidth Threshold for Peer-to-Peer Video-on-Demand Scalability. In: Proceedings of the 23rd IEEE International Parallel & Distributed Processing Symposium (IPDPS). (2009).
17. Bouklit, M., Mathieu, F.: BackRank: an alternative for PageRank?. In: Proceedings of WWW '05: Special interest tracks and posters of the 14th international

conference on World Wide Web. pp. 1122–1123 ACM, Chiba, Japan (2005). https://doi.org/http://doi.acm.org/10.1145/1062745.1062899.

18. Caldarelli, G., Capocci, A.: Beauty and distance in the stable marriage problem. Physica A Statistical Mechanics and its Applications. 300, 325–331 (2001).
19. Castro, M. et al.: Splitstream: High-bandwidth multicast in cooperative environments. In: Proc. of the 19th ACM Symp. on Operating Systems Principles (SOSP). (2003).
20. Cechlárová, K., Fleiner, T.: On a generalization of the stable roommates problem. ACM Trans. Algorithms. 1, 1, 143–156 (2005).
21. Chowla, S.: The asymptotic behavior of solutions of difference equations. In: Proceedings of the International Congress of Mathematicians, Vol. I, 377, Amer. Math. Soc. (1950).
22. Clarke, I. et al.: Freenet: A Distributed Anonymous Information Storage and Retrieval System. Lecture Notes in Computer Science. 2009, 46–66 (2001).
23. Cohen, B.: Incentives Build Robustness in BitTorrent. In: P2PECON. (2003).
24. Diamantoudi, E. et al.: Random paths to stability in the roommate problem. Games and Economic Behavior. 48, 1, 18–28 (2004).
25. Dijkstra, E.W.: Self-stabilizing systems in spite of distributed control. Commun. ACM. 17, 11, 643–644 (1974). https://doi.org/http://doi.acm.org/10.1145/361179.361202.
26. Djungu, S.-J. et al.: SpeedSiteRank : Algorithme parallèle pour un PageRank distribué en sites. In: Proceedings of the 9ème Colloque Africain sur la Recherche en Informatique. (2008).
27. Djungu, S.-J. et al.: Étude comparative des méthodes de calcul de PageRank. In: Proceedings of the 8ème Colloque Africain sur la Recherche en Informatique. (2006).
28. Dumont, D.: Mariages Stables. Séminaire Lotharingien de Combinatoire. 23, (1999).
29. Le Fessant, F. et al.: Clustering in peer-to-peer file sharing workloads. In: IPTPS. (2004).
30. Le Fessant, F., Patarin, S.: Mldonkey, a multi-network peer-to-peer file-sharing program. (2003).
31. Fleiner, T.: The stable b-matching polytope. Mathematical Social Science. 149–158 (2003).
32. Fraigniaud, P., Gauron, P.: D2B: a de Bruijn based content-addressable network. Theor. Comput. Sci. 355, 1, 65–79 (2006). https://doi.org/http://dx.doi.org/10.1016/j.tcs.2005.12.006.

33. Frey, D. et al.: Solipsis: A Decentralized Architecture for Virtual Environments. In: International Workshop on Massively Multiuser Virtual Environments. IEEE (2008).

34. Gai, A., Viennot, L.: Broose: a Practical Distributed Hashtable based on the De-Brujin Topology. In: Proc. of the 4th IEEE Int. Conf. on Peer-to-Peer Computing (P2P). (2004).

35. Gai, A., Viennot, L.: Incentive, Resilience and Load Balancing in Multicasting through Clustered de Bruijn Overlay Network. In: 14th IEEE International Conference On Networks (ICON). (2006).

36. Gai, A.-T. et al.: Acyclic Preference Systems in P2P Networks. In: Proceedings of the 13th International Parallel Processing Conference (Euro-Par). pp. 825–834 (2007).

37. Gai, A.-T. et al.: Stratification in P2P Networks: Application to BitTorrent. In: Proceedings of the 27th IEEE International Conference on Distributed Computing Systems (ICDCS). pp. 30–39 (2007).

38. Gale, D., Shapley, L.: College Admissions and the Stability of Marriage. American Mathematical Monthly. 69, 9–15 (1962).

39. Gaume, B., Mathieu, F.: From Random Graph to Small World by Wandering. (2008).

40. Gaume, B., Mathieu, F.: PageRank-Induced Topology for Real-World Networks. Submitted to Journal of Complex Systems. (2009).

41. Hei, X. et al.: Insights into PPLive: A Measurement Study of a Large-Scale P2P IPTV System. In: In Proc. of IPTV Workshop, International World Wide Web Conference. (2006).

42. Irving, R.W. et al.: An efficient algorithm for the "optimal" stable marriage. J. ACM. 34, 3, 532–543 (1987).

43. Irving, R.W. et al.: The Hospitals/Residents Problem with Ties. In: SWAT '00: Proceedings of the 7th Scandinavian Workshop on Algorithm Theory. pp. 259–271 Springer-Verlag, London, UK (2000).

44. Irving, R.W., Manlove, D.F.: The stable roommates problem with ties. J. Algorithms. 43, 1, 85–105 (2002). https://doi.org/http://dx.doi.org/10.1006/jagm.2002.1219.

45. Iwama, K. et al.: Stable Marriage with Incomplete Lists and Ties. In: ICALP. pp. 443–452 (1999).

46. Kampffmeyer, H. et al.: EPITELLA: Improving the Gnutella Search Algorithm through Epidemic Spreading Models for Complex Networks. In: Proceedings of the IEEE Workshop on Dependable Support in Self-Organising Networks

(DASSON'07). Colocated with 37th Annual IEEE/IFIP Conference on Dependable Systems and Networks. IEEE Press, Edinburgh, United Kingdom (2007).

47. Kawahara, Y. et al.: A Peer-to-Peer Message Exchange Scheme for Large-Scale Networked Virtual Environments. Telecommunication Systems. 25, 3, (2004).
48. Keller, J., Simon, G.: Solipsis: a massively multi-participant virtual world. In: Intern. Conf. on Parallel and Distributed Techniques and Applications. (2003).
49. Kleinberg, J.: The small-world phenomenon: an algorithm perspective. In: STOC '00: Proceedings of the thirty-second annual ACM symposium on Theory of computing. pp. 163–170 ACM, Portland, Oregon, United States (2000). https://doi.org/http://doi.acm.org/10.1145/335305.335325.
50. Lebedev, D. et al.: On Using Matching Theory to Understand P2P Network Design. In: International Network Optimization Conference (INOC). (2007).
51. Lin, Y.-J. et al.: Sync-MS: synchronized messaging service for real-time multi-player distributed games. In: Proc. of the 10th IEEE International Conference on Network Protocols. (2002).
52. Manlove, D.: The structure of stable marriage with indifference. Dis. Appl. Math. 122, 167–181 (2002).
53. Manlove, D.F. et al.: Hard variants of stable marriage. Theor. Comput. Sci. 276, 1–2, 261–279 (2002).
54. Mathieu, F.: Acyclic Preference-Based Systems, (2009).
55. Mathieu, F.: Self-Stabilization in Preference-Based Systems. Peer-to-Peer Networking and Applications. 1, 2, 104–121 (2008).
56. Mathieu, F.: Upper Bounds for Stabilization in Acyclic Preference-Based Systems. In: Proceedings of the Stabilization, Safety, and Security of Distributed Systems, 9th International Symposium (SSS). pp. 372–382 (2007).
57. Mathieu, F. et al.: The Stable Configuration in Acyclic Preference-Based Systems. In: 28th IEEE International Conference on Computer Communications (INFOCOM). (2009).
58. Mathieu, F. et al.: The Stable Configuration in Acyclic Preference-Based Systems. (2008).
59. Mathieu, F., Reynier, J.: Missing Piece Issue and Upload Strategies in Flashcrowds and P2P-assisted Filesharing. In: AICT-ICIW '06: Proceedings of the Advanced Int'l Conference on Telecommunications and Int'l Conference on Internet and Web Applications and Services. p. 112 IEEE Computer Society, Washington, DC, USA (2006).
60. Mathieu, F., Viennot, L.: Local Aspects of the Global Ranking of Web Pages. In: Proceedings of the 6th International Workshop on Innovative Internet Community Systems. (2006).

61. Maymounkov, P., Mazieres, D.: Kademlia: A peer-to-peer information system based on the xor metric. In: 1st International Workshop on Peer-to-Peer Systems (IPTPS). (2002).

62. Qiu, D., Srikant, R.: Modeling and performance analysis of BitTorrent-like peer-to-peer networks. In: Proceedings of ACM SIGCOMM. pp. 367–378 ACM, Portland, Oregon, USA (2004). https://doi.org/http://doi.acm.org/10.1145/1015467.1015508.

63. Qiu, D., Srikant, R.: Modeling and performance analysis of BitTorrent-like Peer-to-Peer networks. In: INFOCOM. (1998).

64. Ratnasamy, S. et al.: A scalable content-addressable network. In: SIGCOMM '01: Proceedings of the 2001 conference on Applications, technologies, architectures, and protocols for computer communications. pp. 161–172 ACM, San Diego, California, United States (2001). https://doi.org/http://doi.acm.org/10.1145/383059.383072.

65. Ronn, E.: On the complexity of stable matchings with and without ties. (1986).

66. Roth, A.E.: The Evolution of the Labor Market for Medical Interns and Residents: A Case Study in Game Theory. Journal of Political Economy. 92, 6, 991–1016 (1984).

67. Roth, A.E. et al.: Pairwise kidney exchange. Journal of Economic Theory. 125, 2, 151–188 (2005).

68. Rowstron, A., Druschel, P.: Pastry: Scalable, Decentralized Object Location, and Routing for Large-Scale Peer-to-Peer Systems. Lecture Notes in Computer Science. 2218, 329–350 (2001).

69. Schuler, F.: Étude et utilisation des technologies des P2P, (2005).

70. Shen, X. et al. eds: Handbook of Peer-to-Peer Networking. Springer (2009).

71. Sripanidkulchai, K. et al.: Efficient Content Location Using Interest-Based Locality in Peer-to-Peer Systems. In: INFOCOM. (2003).

72. Steinmetz, R., Wehrle, K. eds: Peer-to-Peer Systems and Applications. Springer (2005).

73. Stoica, I. et al.: Chord: A scalable peer-to-peer lookup service for internet applications. In: SIGCOMM '01: Proceedings of the 2001 conference on Applications, technologies, architectures, and protocols for computer communications. pp. 149–160 ACM, San Diego, California, United States (2001). https://doi.org/http://doi.acm.org/10.1145/383059.383071.

74. Suh, K. et al.: Push-to-Peer Video-on-Demand System: Design and Evaluation. IEEE Journal on Selected Areas in Communications. 25, 9, 1706–1716 (2008).

75. Sung, Y.-W. et al.: Enabling contribution awareness in an overlay broadcasting system. SIGCOMM Comput. Commun. Rev. 36, 4, 411–422 (2006). https://doi.org/http://doi.acm.org/10.1145/1151659.1159960.

76. Viennot, L. et al.: Il était une fois le peer-to-peer., (2004).

77. Wikibook: The World of Peer-to-Peer (P2P), (2008).

78. Zhang, X. et al.: CoolStreaming/DONet : A Data-driven Overlay Network for Peer-to-Peer Live Media Streaming. In: INFOCOM. (2005).

79. Zhao, B.Y. et al.: Tapestry: An infrastructure for fault-tolerant wide-area location and routing. (2001).

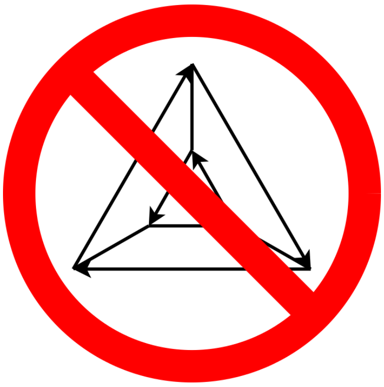
80. Zisk, R.: There's No `I' in Team. Grey's Anatomy. Episode4, (2008).

Résumé

D'un millénaire à l'autre, le pair-à-pair a émergé comme un nouveau paradigme informatique. Plus précisément, des nouveaux enjeux sociaux et économiques, ayant trait en particulier à la distribution de contenus, sont venus raviver d'anciennes problématiques liées aux systèmes décentralisés, leur donnant de nouvelles justifications et de nouveaux éclairages. Dans ce mémoire, je propose tout d'abord de donner des bases pour comprendre et aborder les problématiques pair-à-pair. Après un bref survol des travaux auxquels je me suis intéressé dans le but d'améliorer la distribution de contenu, je me tourne vers un sujet plus théorique : les réseaux à préférences acycliques, lesquels sont un moyen élégant pour modéliser des systèmes pair-à-pair non-structurés ou hybrides. Issus de la théorie des mariages stables, leur principale caractéristique est une capacité auto-stabilisante. Deux questions fondamentales se posent alors, auxquelles je donne un début de réponse : à quelle vitesse un réseau à préférences acycliques se stabilise-t-il, et vers quel état converge-t-il ?

Mots clés

P2P — Théorie des mariages stables — Distribution de contenu



Abstract

Peer-to-peer (P2P) recently emerged as a new paradigm in computer science. Due to major economic and social stakes, mostly related to content distribution, P2P has brought back to the forefront many existing research fields related to distributed systems, providing new incentives and goals. In this work, we give some keys to the understanding of the research fields related to P2P systems. After a brief survey of our work on content distribution, we consider a more theoretical subject: acyclic preference-based systems, which recently appeared as an elegant way to model many P2P unstructured or hybrid systems. The strength of these models is a self-stabilizing property that allows us to provide analytical results in addition to empirical validation.

Keywords

Peer-to-Peer — Stable Marriages — Content Distribution