Stratification in P2P Networks - Application to BitTorrent

Anh-Tuan Gai, Fabien Mathieu, Julien Reynier, Fabien de Montgolfier

▶ To cite this version:

Anh-Tuan Gai, Fabien Mathieu, Julien Reynier, Fabien de Montgolfier. Stratification in P2P Networks - Application to BitTorrent. [Research Report] RR-6081, INRIA. 2006, pp.19. inria-00121974v2

HAL Id: inria-00121974

https://hal.inria.fr/inria-00121974v2

Submitted on 27 Dec 2006

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Stratification in P2P Networks Application to BitTorrent

Anh-Tuan Gai — Fabien Mathieu — Julien Reynier — Fabien de Montgolfier

N° 6081

December 2006

Thème COM

Rapport de recherche

ISSN 0249-6399 ISRN INRIA/RR--6081--FR+ENG

INRIA
ROCQUENCOURT

Stratification in P2P Networks
Application to BitTorrent

Anh-Tuan Gai* , Fabien Mathieu† , Julien Reynier‡ , Fabien de Montgolfier§

Thème COM — Systèmes communicants
Projets Gyroweb

Rapport de recherche n° 6081 — December 2006 — 19 pages

Abstract: We introduce a model for decentralized networks with collaborating peers. The model is based on the stable matching theory which is applied to systems with a global ranking utility function. We consider the dynamics of peers searching for efficient collaborators and we prove that a unique stable solution exists. We prove that the system converges towards the stable solution and analyze its speed of convergence. We also study the stratification properties of the model, both when all collaborations are possible and for random possible collaborations. We present the corresponding fluid limit on the choice of collaborators in the random case.

As a practical example, we study the BitTorrent Tit-for-Tat policy. For this system, our model provides an interesting insight on peer download rates and a possible way to optimize peer strategy.

Key-words: P2P, stable marriage theory, rational choice theory, collaborative systems, BitTorrent, overlay network, matchings, graph theory

CRC MARDI

* INRIA, domaine de Voluceaux, 78153 Le Chesnay cedex, France
† France Telecom R&D, 38–40, rue du général Leclerc, 92130 Issy les Moulineaux, France
‡ LIENS, 45 rue d'Ulm, 75230 Paris Cedex 05, France
§ LIAFA, 175, rue du Chevaleret, 75013 Paris France

Unité de recherche INRIA Rocquencourt
Domaine de Voluceau, Rocquencourt, BP 105, 78153 Le Chesnay Cedex (France)
Téléphone : +33 1 39 63 55 11 — Télécopie : +33 1 39 63 53 30

Stratification dans les Réseaux Pair-à-Pair
Application à BitTorrent

Résumé : Cet article vise à introduire un nouveau modèle d'analyse des réseaux décentralisés basés sur des collaborations entre pairs. Ce modèle repose sur la théorie des mariages stables appliquée à des systèmes possédant une fonction d'utilité globale. Nous étudions la dynamique induite par la recherche pour chacun des meilleurs partenaires possibles et montrons un théorème d'existence–unicité. Nous observons une rapide vitesse de convergence et étudions le phénomène de stratification dans la solution stable, dans le cas où le graphe des collaborations réalisables est complet et dans celui où il est aléatoire. Pour le cas aléatoire, nous présentons une limite fluide de la solution.

Comme exemple pratique, nous étudions la politique *donnant–donnant* employée dans le logiciel de partage *BitTorrent.* Pour ce système, notre modèle fournit des intuitions pertinentes sur les vitesses de téléchargements ainsi que sur les possibilités d'optimisation des paramètres.

Mots-clés : pair-à-pair, mariages stables, choix rationnels, systèmes collaboratifs, BitTorrent, réseaux overlay, couplages, graphes

1 Introduction

Motivation Collaboration-based distributed applications are successfully applied to large scale systems. A system is said to be collaborative when participating peers collaborate in order to reach their own goal (including being altruistic). Apart from well-known content distribution applications [4, 5], collaborating can be applied to numerous applications such as distributed computing, online gaming, or cooperative backup. The common property of such systems is that participating peer exchange resources. The underlying mechanism provided by protocols for such applications consists in selecting which peers to collaborate with to maximize one's peer benefit with regards to its personal interest. This mechanism generally uses a utility function taking local information as input. One can ask if this approach can provide desirable properties of collaboration-based content distribution protocols like scalability and reliability.

To achieve these properties, the famous protocol BitTorrent [4] implements a Tit-for-Tat (TFT) exchange policy. More precisely, each node *knows* a subset of all other nodes of the system and collaborates with the best ones from its point of view: it uploads to the contacts it has most downloaded from in the last 10 seconds. In other words, the utility of peer p for node q is equal to the quantity of data peer q has downloaded from p (in the last measurement period). The main interest in using the TFT policy is the resulting incentive to cooperate. The nature of the utility function then leads to a clustering process which gather peers with similar upload performances together, called *stratification*.

Recently, much research has been devoted to the study of the phenomenon. So far, however, while it has been measured and observed by simulations, it has not been formally proved. Understanding stratification is a first step towards a better comprehension of the impact of the utility function on a system behavior. A theoretical framework to analyze and compare different utility function is needed: choosing a utility function that best suits a given application is quite difficult. More importantly, it is not clear whether the utility functions implemented lead to desirable properties. We introduce a generic framework that allows an instantiation of (known and novel) utility functions that model collaboration. We further present a thorough analysis of a class of utility functions based on global ranking agreements, such as that of BitTorrent TFT policy. This framework also fits gossip-based protocols used by a peer to discover its rank [8].

Contribution First, we propose a model based on the stable matching theory. This model describes decentralized networks where peers rank each others and try to collaborate with the best peers for them.

Second, we focus on systems with a global ranking utility function (each peer has an intrinsic value) in the framework of stable matching. We prove that such a system always admits a unique stable solution towards which it converges. We verify through simulations the speed of convergence without and with *churn* (arrivals ans departures).

Third, we study stratification in a toy model of fully connected networks where every peer can collaborate with all other peers. If every peer tries to collaborate with the same number of peers, we observe disjoint clustering. But with a variable number of collaborations per peer, clustering turns into strong stratification.

Fourth, we describe stratification in random graphs. For Erdös-Rényi graphs, the distribution of collaborating peers has a fluid limit. This limiting distribution shows that stratification is a scalable result.

Lastly, we propose a practical application of our results to the BitTorrent TFT policy. Assuming content availability is not a bottleneck in a BitTorrent swarm, our model leads to an interesting characterization of the download rate a peer can expect as a function of its upload rate. This description leads to possible strategies for optimizing the download for a given upload rate.

Roadmap In Section 2 we define our model. Section 3 presents a study on the problem dynamics. Section 4 describes stratification in a complete neighborhood graph and Section 5, in random graphs. Section 6 discusses the application of our results to BitTorrent and Section 7 concludes the paper.

2 Model

P2P networks are formed by establishing an overlay network between peers. A peer acts both as a server and a client. Each peer p has a bounded number $b(p)$ of collaboration slots. As the network evolves, peers continuously search after new (or better) partners. Each protocol has its own approach to handling these dynamic changes. For example, a protocol like eDonkey [5, 1] optimizes independently two preference lists on the server and on the client sides. More recent protocols, like BitTorrent [4], make a use of a game theoretic approach, where each peer tries to improve its own payoff. It results in keeping one preference list per node.

Let us suppose that each peer p has a *global mark* $S(p)$, which may represent its available bandwidth, its computational capacities, or its shared storage capacity. Each peer wants to collaborate with best partners who have highest marks $S(p)$. This models many networks preferences systems, albeit not all networks have such ranking. For instance, in chess playing, players have an intrinsic value (ELO rating), although they don't generally want to engage people far better or worse than them.

Some peers might not be willing to cooperate with some others. For instance, peers that have no common interest or are unaware of each other. We introduce an *acceptance graph* to represent compatibilities. A pair (p, q) belongs to the acceptance graph if, and only if (iff) both peers are interested in collaboration. Without loss of generality, we can suppose acceptability is a symmetric relation: if p is unacceptable for q, q will never be able to collaborate with p so we can assume q is also unacceptable for p. We denote by *configuration* or *matching* the subgraph of the acceptance graph that represents the effective collaboration between peers. The degree of a peer p in a configuration is bounded by $b(p)$.

A *blocking pair* for a given configuration is a set of two peers unmatched together wishing to be matched together (even if it means dropping one of their current collaborations). A configuration without blocking pair is said to be *stable*. In a stable configuration, a single peer cannot improve its situation: it is a Nash equilibrium.

If a number of colloborations is limited to 1, the problem is known as the *stable roommates problem* [7]. It is an extension of the famous *stable marriage problem* introduced by Gale and Shapley in 1962 [6]. If we assume each peer p wants to collaborate with up to $b(p)$ other peers, the framework is called stable b-matching problem[1] [3].

As it holds for all theories of stable matchings, the existence of a stable configuration depends on the preference rules used to rank participant and on the acceptance graph. In this work we study the impact of the rules derived from a global ranking on a peer-to-peer network behavior. In particular, we find the properties of the stable configurations.

3 Existence and convergence properties of a stable configuration

Global ranking matching is one of the simplest cases of matching problems. Tan [13] has shown that existence and uniqueness of stable solutions were related to preference cycles in the utility function. A preference cycle of length k is a set $i_1, \ldots, i_k$ of k distinct peers such that each peer of the cycle prefers its successor to its predecessor. As proved by Tan, a stable configuration exists iff there is no odd preference cycle of length greater than 1. He also proved that if no even cycle of length greater than 2 exists, then the stable configuration is unique. If peers have an intrinsic value, no strict preferences cycle can occur (see below for ties), so a global ranking matching problem has one and only one stable solution.

This solution is very easy to compute knowing the global ranking S, b and the acceptance graph. The process is given by Algorithm 1: each peer p starts with $b(p)$ available connections. First, the best peer p_1 picks the best $b(p_1)$ peers from its acceptance list. As p_1 is the best, the chosen peers gladly accept (recall the acceptance graph is symmetric) and the resulting collaborations are stable (no blocking pair can unmatch them). Note that if there is not enough acceptable peers, p_1 may not satisfy all its connections. Peers chosen by p_1 have one less connection available. Then second best peer p_2 does the same, and so on... By immediate recurrence, all connections made are stable. When the process reaches the last peer, the connections are the stable configuration for the problem. As it was said before, all connections are not necessarily satisfied. For instance, if the last peer still has available connections when its turn comes, his connections will not be fetched, as all peers above him have by construction spent all their connections. This is, of course, a centralized algorithm, but we shall see below that decentralized algorithms work as well.

Note on ties Ties in preference lists make the matching problems more difficult to resolve [11] without bringing more insight about the stratification issues studied in this paper. Simulations have shown our results hold if we allow ties, but equations are hard to prove as existence of a stable matching cannot be guaranteed. Thus for the sake of simplicity, we shall suppose utilities are distinct, that is $S(q) \neq S(p)$ for any $p \neq q$.

Convergence One can ask what is the point in studying a stable configuration in a dynamical context such as P2P systems, where peers arrive and depart whenever they wish, and where utility functions and acceptance lists can fluctuate. We have not proved yet that the process of peers trying independently to collaborate to the best peers they know can reach the stable state.

[1] in this paper the word *matching* stands for b-matching (unless otherwise stated)

Algorithm 1: Stable configuration in global ranking

```
Data: An acceptance graph $G$ with $n$ peers, a global ranking $S(p)$, and maximal number of connections
      $b(p)$
Result: The unique stable configuration of the $b$-matching problem
Let $a$ be a vector initialized with $b$
for each peer $i$ sorted in increasing $S(p)$ (best peer first) do
    for each peer $j$ sorted in increasing $S(p)$ starting just after $i$ do
        if $(i, j) \in G$ and $a(i) > 0$ and $a(j) > 0$ then
            connect$(i, j)$
            $a(i) = a(i) - 1$
            $a(j) = a(j) - 1$
        end
    end
end
```

We introduce the concept of *initiative* to model the process by which a peer may change its mates. Given a configuration C, we say that peer p *takes the initiative* when it proposes to other peers to be its new mate. Basically, p may propose partnership to any acceptable peer. But only blocking pairs of C represent an interesting new partnership. If p can find such a *blocking mate*, the initiative is called *active* because it succeeds in modifying the configuration (both peers will change their set of mates).

To find a blocking mate, p contacts peers from its acceptance list. We identify several strategies depending on how p scans its acceptance list:

best mate when the peer selects the best (if any) available blocking mate. This happens if p knows the rank of all its acceptable peers and whether they will collaborate or not,

decremental when the list is circularly scanned starting from the last asked peer. This happens if p knows the rank of all its acceptable peers, but not if they will collaborate,

random when a single peer is selected at random. This happens if p has no information on its neighbors until it asks.

Of course, when best mate initiative is possible, it seems to be the best strategy to maximize a peer's own profit, but it supposes a good knowledge of the system is maintained.

We can now complete our model with initiatives: starting from any initial configuration, an instance of our model evolves because of initiatives taken by peers. In fact, it can only evolve towards the unique stable configuration, as shown by Theorem 1.

Theorem 1 *The stable solution can be reached in $B/2$ initiatives, where $B = \sum_p b(p)$ is the maximal number of connections. Moreover, any sequence of active initiatives starting from any initial configuration eventually reaches the stable configuration.*

Proof: In Algorithm 1, each connection can be obtained by initiative. As the stable configuration possesses up to $B/2$ pairings, this ensures the first part of the theorem. We prove the convergence by showing a sequence of active initiatives can never produce twice the same configuration. There is a finite number of possible configurations, so if we keep altering the configuration through initiatives, we eventually reach a configuration that cannot be altered with any initiative: the stable configuration.

The proof is indeed simple. If a sequence of initiatives induces a cycle of at least two distinct configurations, then one can extract a preference cycle of length greater than 3: let p_1 be a peer whose mates change through the cycle. Call p_2 the best peer p_1 is unstably paired with during the cycle, and p_3 the best peer p_2 is unstably paired with during the cycle. p_1 is not p_3 and p_2 prefers p_3 to p_1, otherwise the pair $\{p_1, p_2\}$ would not break during the cycle. Iterating the process, we build a sequence of peer (p_k) such that p_k prefers p_{k+1} to p_{k-1}, until we find $i < j$ such that $p_i = p_j$. The circular list $(p_i, p_{i+1}, \ldots, p_{j-1})$ is a preference cycle. As global ranking does not allow preference cycles, this is not possible, so a sequence of active initiatives can never produce twice the same configuration. □

Theorem 1 proves that in static conditions (no join or departure, constant utility function), a P2P system will converge to the stable state. To prove this stable state is worth studying, we have to show convergence

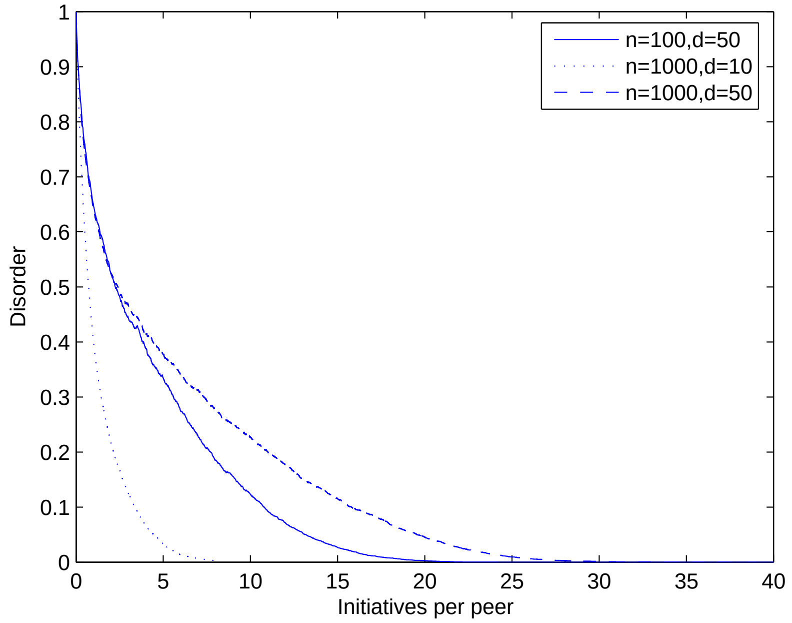


Figure 1: Starting from $C_\emptyset$, convergence towards the stable state for different parameters

is fast in practice (Algorithm 1 is optimal in number of initiatives but difficult to implement in a large scale system) and can sustain a certain amount of churn. As a complete formal proof of this is beyond the scope of this paper, we used simulations.

In our simulations, peers were labeled from 1 to n (the number of peers). These labels define the global ranking, 1 being the best peer and n the worst (if $i < j$, peer i is better than peer j). We use Erdös-Renyi loopless symmetric graphs $G(n, d)$ as acceptance graphs, where d is the expected degree (each edge exists independently with probability $\frac{d}{n-1}$). Only 1-matching was considered.

For measuring the difference between two configurations C_1 and C_2 we use the distance

$$D(C_1, C_2) = \Sigma_{i=1}^{n} \|\sigma(C_1, i) - \sigma(C_2, i)\|.\frac{2}{n(n+1)},$$

where $\sigma(C, i)$ denotes the mate of i in C (by convention, $\sigma(C, i) = n + 1$ if i is unmated in C).

D is normalized: the distance between a complete matching and the empty configuration $C_\emptyset$ is equal to 1. The *disorder* denotes the distance between the current configuration and the stable configuration.

At each step of the process we simulate, a peer is chosen at random and performs a best mate initiative (the initiative can be active or not). To compare simulations with different number n of peers, we take a sequence of n successive initiatives as a base unit (that can be seen as *one expected initiative per peer*).

A first set of simulations is made to prove a rapid convergence when the acceptance graph is static. In all simulations, the disorder quickly decreases, and the stable configuration is reached in less than nd initiatives (that is d base unit). Figure 1 shows convergence starting from the empty configuration for three typical parameters: $(n, d) = (100, 50)$, $(n, d) = (1000, 10)$, $(n, d) = (1000, 50)$.

Then we investigate the impact of an atomic alteration of the system. Starting from the stable configuration, we remove a peer from the system and observe the convergence towards the new stable configuration. We observe big variances in convergence patterns, but convergence always takes less than d base units and disorder is always small. Note, that due to a domino effect, removing a good peer generally induces more disorder than removing a bad peer. This is shown by Figure 2. We ran the simulations 100 times and selected four representative trajectories, as we did not wish to average out interesting patterns.

Finally, we investigate continuous churn. A peer can be removed or introduced in the system anytime, according to a *churn rate* parameter. Simulations show that as the churn rate increases, the system becomes unable to reach the instant stable configuration. However, the disorder is kept under control. That means

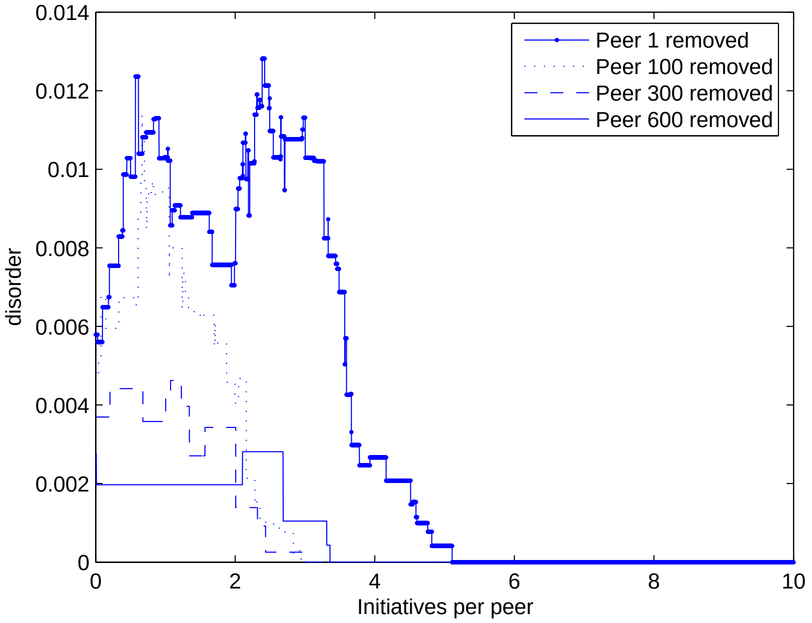


Figure 2: Starting from the stable state, we remove a peer and observe the convergence towards the new stable state. (1000 users, 1-matching, 10 neighbors per peer)

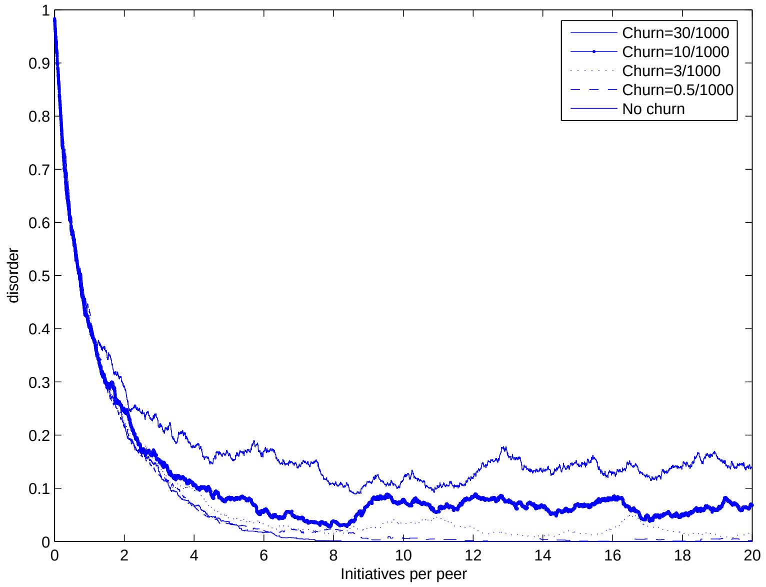


Figure 3: Starting from $C_\emptyset$, we observe distance to the instant stable state with different churn levels (1000 users, 1-matching, 10 neighbors per peer)

the current configuration is never far from the instant stable configuration. The average disorder is roughly proportional to the churn rate (see Figure 3 for typical patterns).

All these simulations lead to the same idea: the stable configuration acts like a strong attractor in the space of possible configurations when collaborations are established using intrinsic values for judging peers Studying the properties of the stable configuration is the next step.

4 Stratification with complete acceptance graph

We start studying the stable configuration in the special case where everybody is acceptable for everybody. Hence the acceptance graph is complete. This is a suitable, but not scalable, assumption for small systems. Complete acceptance graph is a toy model for highlighting stratification effect.

4.1 Clustering in constant b-matching

Constant b-matching is an instance of the b-matching problem where every one tries to connect to at most b_0 peers (b_0 is a constant). Since the acceptance graph is complete, the stable configuration is very simple. It consists in a sequence of complete subgraphs with $b_0 + 1$ elements starting from the best peer (the remainder, if any, is a truncated complete subgraph). For example, figure 4 shows this *clustering* for the 2-matching problem on a complete graph.

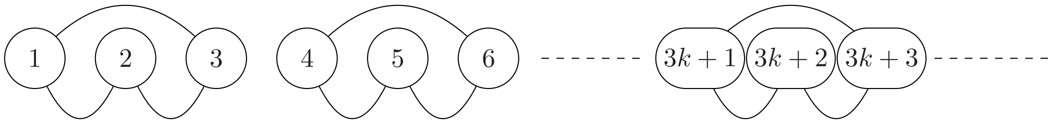


Figure 4: Limit case of b-matching and total knowledge : the collaboration graph is a set of $b+1$ clusters. Here $b = 2$.

As it has already been pointed out [2], full clustering in file sharing networks induces poor performances. Many designers try to produce overlay graphs with small world properties: almost fully connected, high clustering coefficient, low mean distance, and navigable such that shortest paths may be greedily found. But in file sharing networks, having a compliant overlay with nice properties (connectivity, distances, resilience) is useless if the effective collaborations graph has none of the desired properties. In our example, although the knowledge graph is a complete graph, collaboration established through global ranking scatters the graph in clusters. Hence content is sealed inside clusters, and singularities are bound to occur.

Lower bound for number of slots in BitTorrent

As we have just spoken of clustering, it is interesting to remember that a connected graph of n vertex has at least $n - 1$ edges. As a b_0-regular graph has $\frac{b_0}{2}n$ edges, it is impossible for a 1-regular graph to be connected, and the cycle is the unique 2-regular connected graph. It follows that it is better to set $b_0 \geq 3$.

This gives a first basic insight for the fact that the default number of slots per user is 4 is BT (less for very small connections and more for high bandwidth ones): given the generous extra slot, put less than 4 slots in the default client would make the TFT collaboration graph disconnected which would seriously harm the BT efficiency.

Of course, BT is more complicated, and this is just a by-passing remark. In Section 6 we propose further arguments to see why 4 seems to be the number of connections the average client should set by default.

4.2 Stratification in variable b-matching

b_0-matching is not the most common case in practice. The clustering from Figure 4 may be a consequence of the specific parameters used. Indeed, adding only one connection can alter a set of complete subgraphs of size $b_0 + 1$ in one unique connected component (see Figure 5 – settings are same than for Figure 4 except that an extra connexion has been granted to peer 1).

In fact, both Figures 4 and 5 are not typical. In our simulations on complete acceptance graphs, we generally observed many large connected components. If we assume that b is distributed according to a rounded normal distribution $\mathcal{N}(\bar{b}, \sigma^2)$ (mean $\bar{b}$, variance σ, all samples are rounded to the nearest positive integer), we observe a surprising phase transition. As soon σ is big enough to produce heterogeneous samples ($\sigma \approx 0.15$), the average

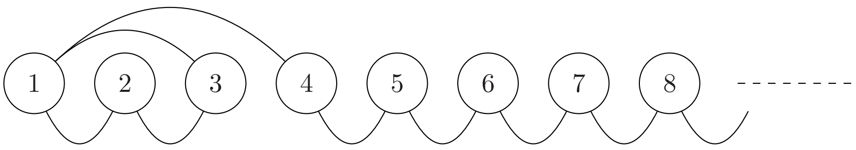


Figure 5: b-matching plus one extra connection: the graph is connected

	constant b_0-matching						normal $\mathcal{N}(b,\sigma^2)$-matching with $\sigma = 0.2$					
b_0 or $\bar{b}$	2	3	4	5	6	7	2	3	4	5	6	7
Average Cluster Size	3	4	5	6	7	8	6	20	78	350	1800	11000
Max Mean Offset (MMO)	1.67	2.5	3.2	4	4.71	5.5	1.33	2.10	2.52	3.21	3.65	4.31

Table 1: Clustering and stratification properties in a complete knowledge graph.

connected component size explodes, then stays almost constant. The cluster typical size after the transition seems to grow factorially with $\bar{b}$ (Figure 6 shows what happens for $\bar{b} = 6$). Computed values appear in Table 1.

Factorial cluster size growth grants the existence of a giant connected component when $\bar{b}$ is large and n remains bounded. This solves the clustering issue.

Nevertheless, distances in the obtained collaboration graph are another question. A good estimate is given by Mean Max Offset (MMO) which described the mean ranking offset between one peer and its further neighbor in the collaboration graph. The larger the MMO, the fewer hops needed to link two peers with very different intrinsic value in the same connected component. Remark that in b_0-matching, MMO is easy to compute (it is enough to compute it on the $b_0 + 1$ complete graph). We show that it converges to:

$$\begin{aligned} MMO(b_0) &= \tfrac{1}{b_0+1}(b_0 + (b_0 - 1) + \ldots + \lceil \tfrac{b_0}{2} \rceil + \ldots + b_0) \\ &\xrightarrow[b_0 \to +\infty]{} \tfrac{3}{4} b_0. \end{aligned}$$

When b is variable, MMO becomes less obvious to compute. However, simulations show that MMO reflects the same phase transition as the cluster size does. In contrast, as cluster size explodes, MMO decreases, has shown by Figure 6 and Table 1.

The conclusion of this first approach on complete graphs is that whereas the clustering problem can be handled, a stratification issue exists: peers only collaborate with very close to them, which can make content diffusion ineffective.

5 Global ranking on random acceptance graphs

In this section, for the sake of simplicity, we first describe a 1-matching model. This allows us to explain our *independence* assumption and to present the related mathematical results. We then extend the equations to the b_0-matching case, for any constant number b_0 of connexions.

5.1 Model

As noted in Section 3, there exists a unique matching (stable configuration) where no peer can locally improve its mates among its known peers; this matching, denoted P in the following, can be obtained simply by applying algorithm 1. As we first focus on 1-matching, we denote $P(i)$ the mate of Peer i in P.

5.1.1 An exact formula

Denote by $D(i,j)$ the probability that Peer i is matched with Peer j over all possible graphs with n vertices. In other words $D(i,.)$ is the distribution of the peer matched with i.

Obviously, $D(i,j) = D(j,i)$ and $D(i,i) = 0$. The total order property can be written as follows, for $i < j$:

$D(i,j) = p\mathbb{P}($"i is not with better than j" and "j is not with better than i"$)$ where p is the probability that peer i knows peer j.

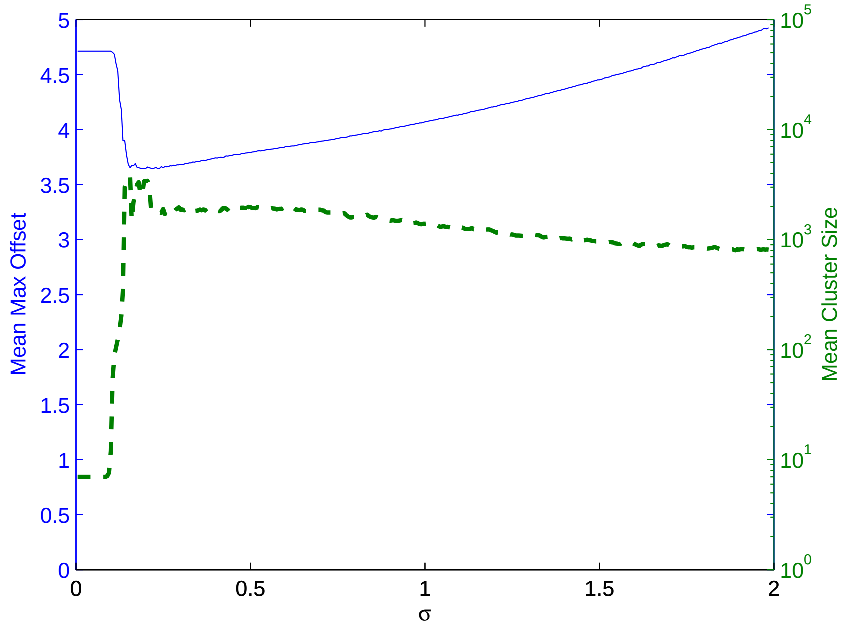


Figure 6: Influence of σ for the b-matching global ranking problem when b follow a normal law $\mathcal{N}(6, \sigma)$. The dotted line represents the Mean Cluster Size, the plain line stands for Mean Max Offset. The left part (when $\sigma = 0$) is the case of constant 6-matching

We can rewrite the above probability as: $\mathbb{P}(P(i) \geq j) \times \mathbb{P}(P(j) \geq i | P(i) \geq j)$, leading to the exact formula:

$$D(i,j) = p(1 - \sum_{k=1}^{j-1} D(i,k))\mathbb{P}(P(j) \geq i | P(i) \geq j). \tag{1}$$

Note that this does not depend on the number of peers. The formula can thus be extended to every couple $(i,j) \in (\mathbb{N}^*)^2$.

Lemma 1

$$\forall i \in \mathbb{N}^*, \sum_{k=1}^{\infty} D(i,k) = 1.$$

This lemma means that, under the Erdös-Rényi assumption, when adding a large number of peers at a lower rank, any peer will eventually find a mate with probability one.

Proof: The conditional probability does not go to 0**:** We first show that $\mathbb{P}(P(j) \geq i | P(i) \geq j)$ does not go to 0. Suppose that $j > i$, then condition on $E_i = \{P(1), ..., P(i-1)\}$,

$$\mathbb{P}(P(j) \geq i | P(i) \geq j | E_i) = \begin{cases} \text{empty conditioning if } i \in E_i \\ 0 \text{ if } j \in E_i (\text{and } i \notin E_i) \\ x \geq p \text{ if } j \notin E_i \text{ and } i \notin E_i. \end{cases}$$

The last inequality holds because if $j \notin E_i$ and $i \notin E_i$, then knowing that $P(i) \geq j$, i and j are linked if and only if there exists an edge between both. Since $P(j) = i$ implies as a particular consequence $P(j) \geq i$, the inequality is satisfied.

Now, all we have to show is that $\mathbb{P}(j \in E_i | i \notin E_i)$ does not tend to 1 when j tends to infinity. This is obvious since for some $k < i$, the function $j \to \mathbb{P}(P(k) = j | i \notin E_i)$ gives probabilities of disjoint events so that $\sum_{j=1}^{\infty} \mathbb{P}(j \in E_i | i \notin E_i) \leq i - 1$; the general term thus tends to 0 and certainly not to 1.

D is a probability We know that for a given i, $D(i,j)$ are the probabilities of disjoint events. Thus $D(i,j) \xrightarrow[j\to\infty]{} 0$. From formula (1) we deduce

$$\sum_{k=1}^{j-1} D(i,k) \xrightarrow[j\to\infty]{} 1.$$

□

5.1.2 Approximation: independent 1-matching model

Hereinafter we shall adopt the following assumption:

Assumption 1 *the two events:*

- *peer i is not with a peer better than j,*
- *peer j is not with a peer better than i,*

are independent.

Assumption 1 is reasonable when the probability that i and j have a common neighbor is very low. It entails that (1) can be replaced by the approximate recurrence relation:

$$D(i,j) = p\left(1 - \sum_{k=1}^{j-1} D(i,k)\right)\left(1 - \sum_{k=1}^{i-1} D(j,k)\right) \tag{2}$$

This formula can easily be computed in an iterative way by calculating for increasing i the probabilities $D(i,j)$ from $j = 1$ to n using Algorithm 2 (see Algorithm 3 for the b_0-matching case).

Algorithm 2: Independent 1-matching probability computation

```
Data: Number of peers, $n$
Erdös-Rényi probability, $p$
Result: $D(i,j)$ the probability user $i$ chooses user $j$
$D \leftarrow zeros(n,n)$
for $i = 1$ to $n$ do
    for $j = i+1$ to $n$ do
        $D(i,j) \leftarrow p\left(1 - \sum_{k=1}^{j-1} D(i,k)\right)\left(1 - \sum_{k=1}^{i-1} D(j,k)\right)$
        $D(j,i) \leftarrow D(i,j)$
    end
end
```

Example where the simplified formula does not work Even if the approximation made by using (2) instead of (1) works very well for small values of p (see figure 9), it in not an exact formula. Example in Figure 7 illustrates this point: we considered 3 peers; then we can write down all the possible graphs (8 of them) with the exact probability for each event.

5.2 Main result on the independent 1-matching model

This section presents mathematical results that follow from assumption 1. When the number of peers is large, the model scales and the normalized histogram of neighbors tends to a continuous distribution and yields an equation satisfied in this limit. Indeed the empirical distribution also converges, which means that every instance of an Erdös-Rényi graph is very likely to behave like the typical case of the above assumption, as shown by the simulations below.

We are able to prove some parts of this program but must leave the remainder as conjectures for further work. The results bring considerable insight.

From a practical point of view one only need to retain two points from the mathematical developments:

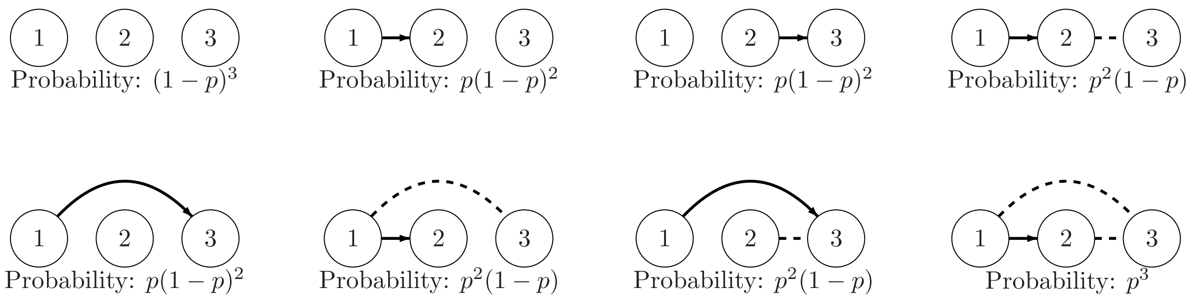


Figure 7: Approximation error: for $n=3$, there is 8 possible graphs. Exact matchings probabilities are
– $D_{\text{exact}}(1,2)=p$
– $D_{\text{exact}}(1,3)=p(1-p)$
– $D_{\text{exact}}(2,3)=p(1-p)^2$
Algorithm 2 leads to the same except

$$\begin{aligned} D(2,3) &= p(1-D(2,1))(1-D(3,1)) \\ &= p(1-p)(1-p(1-p)) \\ &= D_{\text{exact}}(2,3)+p^3(1-p) \end{aligned}$$

- for not moderate values of n, there exists a scaled version of $D(i,j)$ which does not depend on n (see 5.3),
- the shape of $D(i,j)$ is present in almost any given n-peers system.

5.2.1 Distribution weak convergence

Notation, Hypotheses: For all theorems and proofs of this sections, $G=(V,E)$ is an Erdös-Rényi graph, P is its unique stable pairing, and $\mathcal{M}_i(n,p)$ is the distribution of the mate of peer i:

$$\mathcal{M}_i(n,p)=\sum_{j\in[|1,n|]\setminus\{i\}} D(i,j)\delta_j.$$

The mean degree of a peer is denoted d.

Theorem 2 $\mathcal{M}_i(n,p)\xrightarrow[n\to\infty]{*}\mathcal{M}_i(p)$ *with :*

- $\mathcal{M}_i(p)\in\mathcal{P}(\mathbb{Z})$,
- *on the restricted support* $[|1,n|]$*:* $\mathcal{M}_i(n,p)(dx)=\mathcal{M}_i(p)(dx)$.

Theorem 3 (Dirac limit) *We look at the probability* $\mathcal{M}_i(p)(ndx)$ *on the space* $(\frac{1}{n}\mathbb{N},\mathbb{P}^n)$ *where* $\mathbb{P}^n$ *puts probability* 1 *on points of* $\frac{1}{n}\mathbb{N}$*. As a measure on* $\mathbb{R}$*,* $\frac{1}{n}\mathbb{P}^n$ *tends to the Lebesgue measure on* $\mathbb{R}^+$ *for weak convergence; we thus have our first scaling: given p and* $n\to\infty$*:*

$$\forall p, \mathcal{M}_i(p)(ndx)\xrightarrow[n\to\infty]{*}\delta_0.$$

Conjecture 1 (Fluid limit) $n\to\infty$*,* $p^n=\frac{d}{n}$ *consider peer number* $i^n=1+\lfloor n\alpha\rfloor$*; then there exists* $\mathcal{M}_{\alpha,d}\in\mathcal{P}(\mathbb{R})$ *that is absolutely continuous with respect to Lebesgue measure such that:*

$$\mu^n_{\alpha,d}:=\mathcal{M}_{i^n}(p^n)(ndx)\xrightarrow[n\to\infty]{*}\mathcal{M}_{\alpha,d}.$$

Proof of Theorem 2

Theorem 2 is obvious except for the fact that $\mathcal{M}_i$ has mass 1; the result essentially comes from the fact that the probability for peer i to be matched with peer j does not depend on peers with rank greater than the maximum of i and j. Thus the distribution for n peers is only a cut version of the distribution with more peers.

Now we shall prove that the mass is equal to 1. We already know that the mass is equal to 1 for the exact model (Lemma 1). However, it is not obvious this is still true after the changes we made to the toy model. The fact that $\mathcal{M}_i(n,p) \to \mathcal{M}_i(p)$ gives the mass as an increasing limit. First, suppose the mass $\mathcal{M}_i(n,p)$ does not tend to 1. Then there exists some $\epsilon > 0$, such that $\sum_{k=1}^{\infty} D(i,k) < 1 - \epsilon$. If we put this back in formula (2), then:

$$D(i,j) \geq p\epsilon \left(1 - \sum_{k=1}^{i-1} D(k,j)\right) \tag{3}$$

We know that $(D(i,j))_{j=1..\infty}$ is a sub probability, thus $D(i,j) \to 0$ when $j \to \infty$. From equation (3) it follows that $\sum_{k=1}^{i-1} D(k,j) \to 1$. A particular consequence is that for j large enough (i.e., there exist $j_0, such that for all j \geq j_0$), we have: $\sum_{k=1}^{i-1} D(k,j) \geq \frac{1}{2}$ but this is impossible since the $i-1$ sequences $(D(k,j))_{1 \leq k < i; j=1..\infty}$ are probabilities.

Proof of Theorem 3

The result is obvious since all the mass stays in compact sets (tightness property on the Polish space $\mathbb{R}$) and $\mathcal{M}_i(p)$ is a probability. But the fact is interesting for its physical interpretation.

Sketch of proof of Conjecture 1

This is a very technical result. We will only address here the special case where $\alpha = 0$. From a technical point of view, we first have to prove that the sequence μ^n is tight, which allows us to extract a limit. We then have to show that this limit is unique.

In the special case $\alpha = 0$, let $\beta \in \mathbb{R}$ and $j^n = 1 + \beta n \lfloor n \rfloor$ then: $D(1, j^n) = p^n (1 - p^n)^{i^n - 1}$. This implies

$$nD(1, j^n) \sim d \left(1 - \frac{d\beta}{n\beta}\right)^{n\beta} \to de^{-\beta d}.$$

This in turn yields:

$$\mathcal{M}_{0,d}(\mathrm{d}\beta) = de^{-\beta d}\mathrm{d}\beta.$$

This theoretical result could be proven though at the expense of very long and technical developments. We do not anticipate any significant mathematical difficulty though it does remain to carry through the demonstrations. The results are not necessary to make the following observations, but they explain why we have considered some particular scalings.

5.3 Observations

The results in this section are obtained by solving Equation 2. We took $n = 5000$ to obtain the smoothest possible curves but $n = 100$ would give pretty similar results. In Figure 8 we illustrate the different cases that may arise.

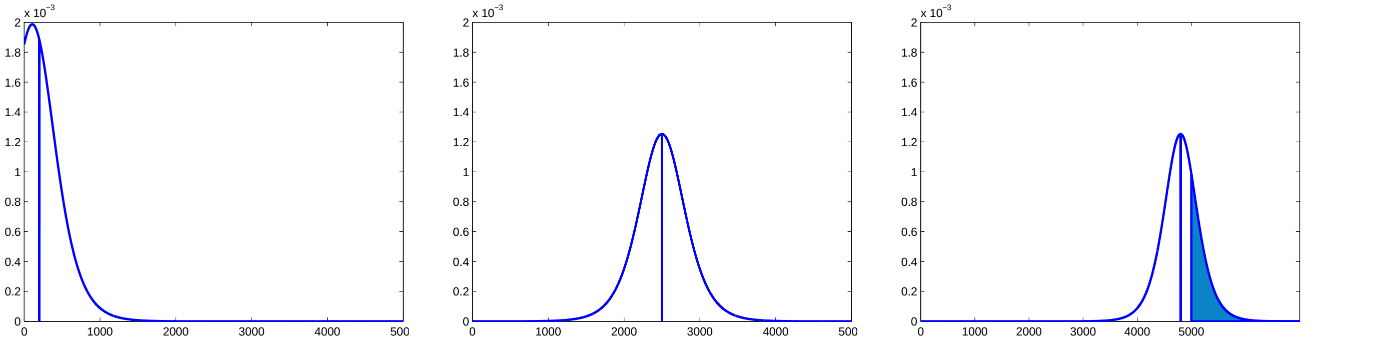
In Figure 8(a) we see the case of a well ranked peer. Note that for $i = 1$ the right part is almost geometrically distributed. Also note that the best peers are peered with peers of lower average rank, but that this changes quickly and peers in the top 20% but not in the top 5% have a significantly better mate on average.

The central case is illustrated in Figure 8(b). We see that the distribution is symmetric and that the distribution simply shifts with the rank of the peer (for top 25% to top 80% peers). This second fact is a kind of finite horizon property and illustrates the property we called stratification. Notice that the distribution can not be fit with a normal law, in any case.

In Figure 8(c), the distribution shift continues for the bottom 20% of peers, but as there is no worse peer to mate with, the distribution is cut. This means that there is a probability for not being matched which is given by the area filled in blue. A particular case for the worst peer is that it will be matched exactly in half of the cases. All the others are assured to do better in terms of matching frequency.

5.4 b_0-matching independent model

The 1-matching case was only presented to give a flavor of the stratification phenomenon. Formally there are no new issues in progressing to a b_0-matching model except for the weight of notation. As for 1-matching, we state an independence assumption which is not formally true but supplies a fairly good approximation compared to simulations as shown in paragraph 5.4.3.



(a) $D(200, j)$

(b) $D(2500, j)$

(c) $D(4800, j)$

Figure 8: Distribution of neighbors in independent 1-matching for $n = 5000$ peers and $p = 0.5\%$ for Peer 200, Peer 2500 and Peer 4800.

5.4.1 Notation

n still denotes the number of peers. The situation becomes more complicated, because the first choice of one peer may correspond to the last choice of its mate. Consequently we have to study a quantity $D_{ci}^{cj}(i, j)$ which is not of directly interest. This is the probability that choice number ci of Peer i is j and that for j, i is choice number cj. As in the 1-matching case, $D_{ci}^{cj}(i, j)$ does not depend on larger indexes for i, j, ci and cj. Nor does it depend on n. Intuitively this corresponds to the fact that the first choice is made before making the second, and that the best peers have priority for choosing their mates. The quantity of interest is $D_{ci}(i, j) = \sum_{cj=1}^{b_0} D_{ci}^{cj}(i, j)$.

Assumption 2 *Let $i, j \leq n$ and $c_i \geq 1$ and $cj \geq 1$, the events:*

- *peer i has chosen $ci - 1$ peers better than j and choice ci is not matched by better than j,*
- *peer j has chosen $cj - 1$ peers better than i and choice cj is not matched by better than i,*

are independent.

The way to evaluate $D_{ci}^{cj}(i, j)$ is to multiply the probabilities of the supposed independent events:

- i knows j: with probability p,
- choice ci of i is not matched and previous choices are matched with better than j,
- the reciprocal condition on j.

Note that the probability that choice ci of i is not matched and previous choices are matched with better than j is simply: $\sum_{k=1}^{i-1} D_{ci-1}(i, k) - \sum_{k=1}^{i-1} D_{ci}(i, k)$, the probability that choice $ci - 1$ is matched with better than j minus the probability that choice ci is matched with better than j (mathematically this formula is exact because one of the two events is included in the other).

This proves, under assumption 2, that:

$$D_{ci}^{cj}(i, j) = p \left(\sum_{k=1}^{j-1} D_{cj-1}(j, k) - D_{cj}(j, k) \right) \left(\sum_{k=1}^{i-1} D_{ci-1}(i, k) - D_{ci}(i, k) \right). \qquad (4)$$

We now show how to compute this formula by recurrence.

5.4.2 Independent b_0-matching algorithm

Note in the following algorithm that $D_c(i, j)$, the c-th choice distribution of i is no longer symmetric for $c > 1$, but $D_{ci}^{cj}(i, j)$ has more symmetry (see Algorithm 3). Matlab scripts can be found at [10]. This version is not optimized (but sufficiently difficult not to do so); the partial sums can be kept in memory to gain a linear factor.

Algorithm 3: Independent b_0-matching probability computation

```
Data: Number of peers, $n$
      Erdös-Rényi probability, $p$
      Number of matchings, $b_0$
Result: $D_{ci}^{cj}(i,j)$ the probability that the $ci$-th choice of Peer $i$ is $j$ and that the $cj$-th choice of $j$ is $i$ and
        $D_c(i,j)$ the probability that the $c$-th choice of Peer $i$ is $j$
$D_c \leftarrow zeros(b_0, n, n)$
$D_{ci}^{cj} \leftarrow zeros(b_0, b_0, n, n)$
$D_0^c \leftarrow ones(1, b_0, n, n)$
$D_c^0 \leftarrow ones(b_0, 1, n, n)$
for $i = 1$ to $n$ do
    for $j = i + 1$ to $n$ do
        for $(ci, cj) \in [|1, b_0|] \times [|1, b_0|]$ do
            $$D_{ci}^{cj}(i,j) \quad \leftarrow \quad p\left(\sum_{k=1}^{j-1} D_{cj-1}(j,k) - D_{cj}(j,k)\right)\left(\sum_{k=1}^{i-1} D_{ci-1}(i,k) - D_{ci}(i,k)\right)$$
        end
        for $ci = 1$ to $b_0$ do
            $D_{ci}(i,j) \leftarrow \sum_{cj=1}^{b_0} D_{ci}^{cj}(i,j)$
        end
        for $cj = 1$ to $b_0$ do
            $D_{cj}(j,i) \leftarrow \sum_{ci=1}^{b_0} D_{ci}^{cj}(i,j)$
        end
    end
end
```

5.4.3 Validation of independent b_0-matching

As mentioned above, assumptions 1 and 2 work very well except for very small numbers of peers with p very large. Figure 9 illustrates this point. We simulated a 2-matching by drawing a million realizations of the Erdös-Rényi graph with $n = 5000$ and $p = 1\%$ (simulations requiring several weeks) and compared distributions $D_1(3000, j)$ and $D_2(3000, j)$ with those given by our simplified formula. The comparison in Figure 9 illustrates the accuracy of the formula.

6 Application to BitTorrent

Results of previous Sections allow us to closely estimate for each peer the ranks of peers it is likely to collaborate with. All our results tend to give a theoretical proof of the stratification phenomenon in systems that use a global ranking function that is not correlated to the acceptance graph. In this Section, we will see how this stratification can give insight about the effect of the Tit-for-Tat policy used in BitTorrent.

We suppose that we are in the *post flashcrowd* phase. In the *flashcrowd* phase, an unique seed is uploading a new file, and the upload capacities of the best peers are useless: all peers have downloaded the same blocks. But during the post flash crowd phase, all blocks have roughly the same repartition, because of the download rarest first policy of BitTorrent. So we can assume content availability will not affect the acceptance graph and focus on bandwidth only.

The TFT policy consists in uploading to the peers from which one gets the best download rates. The selection process is renewed periodically. Along with a generous upload connection that allows to probe new peers for an eventual TFT exchange, this acts like the random peer initiative described Section 2. This why we claim our results apply to the TFT exchanges in BitTorrent. In peculiar, we have a proof of the stratification effects (peers tend to exchange with peers with similar bandwidths) empirically observed by [2, 9].

However, the ranking of a peer just gives an intuition about the Quality of Service (QoS) it is presumed to experience. In order to obtain relevant results, it is therefore necessary to bind ranking and performance. In the case of a file sharing system like BitTorrent, the average expected download rate is a very convenient

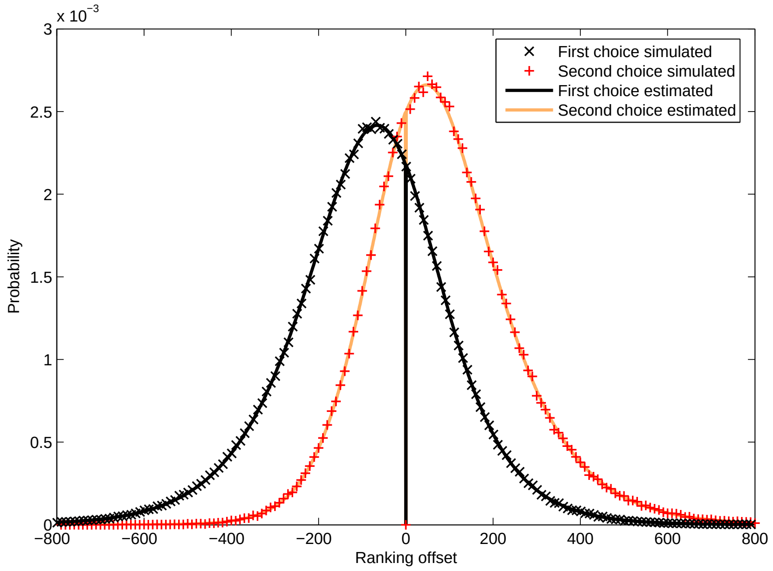


Figure 9: Comparison for $n = 5000$ and $p = 1\%$ (which gives in average 50 neighbors per peer) of the distributions D_1 and D_2 for peer 3000 centered at 3000 with the statistics obtained by simulating without any approximation many instances of Erdös-Rényi random graphs.

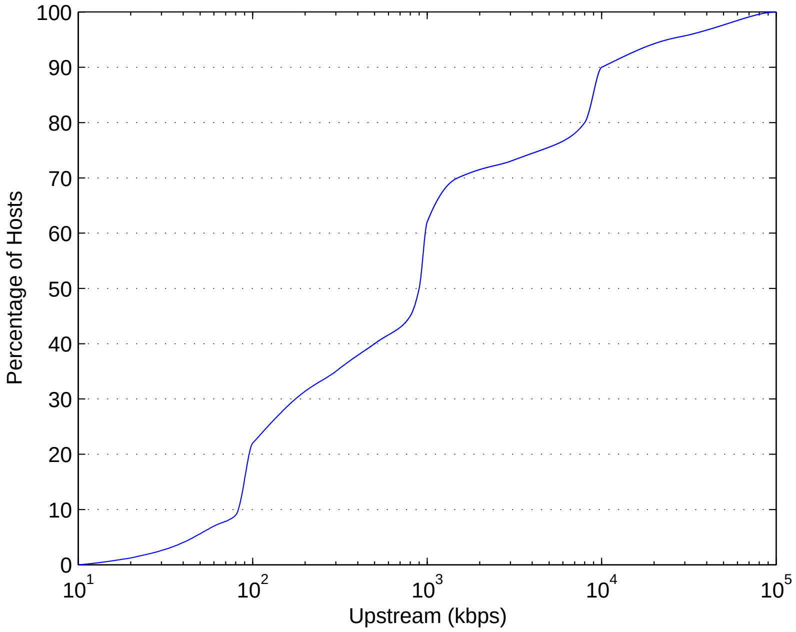


Figure 10: Estimation of bandwidth capacities derived from [12]

performance metric all the more so since it is easy to compute within our model: it is enough to know the upload bandwidth for each peer i.

To compute network performances, we have taken as reference the measurements made by Saroiu *et al.* [12]. Using bandwidth estimation in the Gnutella network, they have estimated the upstream for a large community of P2P users. The cumulative distribution they obtained is shown Figure 10. One can observe a wide distribution of bandwidths (just like in Orwell's *Animal Farm*, "all peers are equal but some peers are more equal than others").

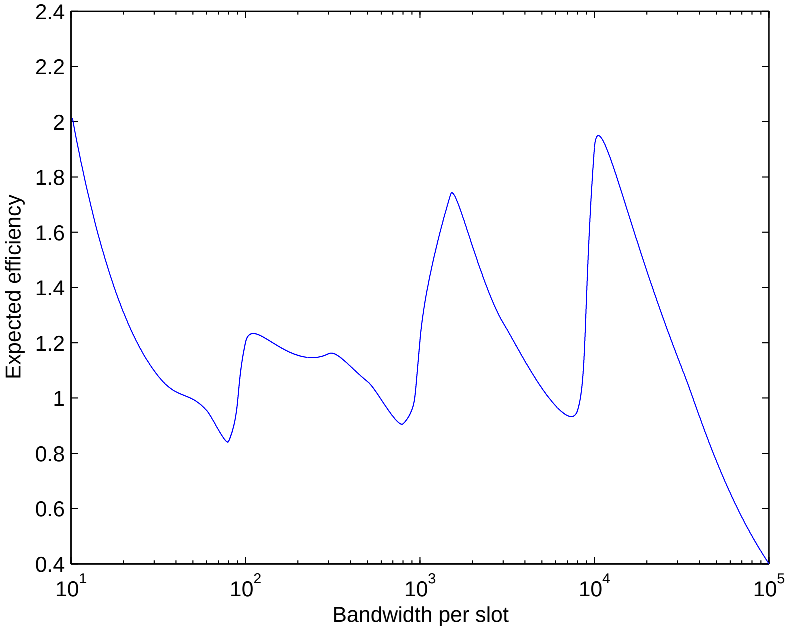


Figure 11: Expected D/U ratio as a function of the upload bandwidth offered. b_0 is set to 3 and average number of neighbors is 20.

Applying our model to the distribution observed by Saroiu *et al.*, we get the results shown in Figure 11. We chose the following parameters:

- b_0-matching with $b_0 = 3$, corresponding in a BitTorrent network with all clients having the default number of slots of 4.
- expected number of acceptable peers (peers who are known and interesting) $d = 20$ (realistic value)

Notice that the number n of peers does not have to be given because our model does not depend on the network size: with a partial network knowledge, observed offsets scale with the number of peers (see Section 5.1).

To put results in the clearest possible way, we chose to represent expected download/upload ratio, which correspond to BitTorrent share ratio. When this ratio is lesser than 1, one gives in average more that it receives.

Some observations are worth being said:

- Best peers suffer from low sharing ratios: as they are the best, they can only collaborate with lower peers, so the exchange is suboptimal for them. The only way for best peers to counter this effect is by adding extra connections until the upload bandwidth per slot is close to the one of lower peers. This somehow explains why BitTorrent proposes *by default* a greater number of connections (up to TCP limitations) for peers with high bandwidths, thus avoiding too much spoil.
- There is density peeks in the bandwidth distribution. this peeks corresponds to typical Internet connections, such as DSL or cable. Peers in the density peeks have a ratio close to 1. This is due to the great probability they have to collaborate with peers that have exactly the same characteristics as them.
- Efficiency peeks appear for peers that have an upload just above a density peek. For these peers, lower peers have almost the same upload bandwidth as them, whereas upper peers are likely to offer greater bandwidth.
- Surprisingly, the lowest peers have a high efficiency, although there is some probability for them not to be matched, as pointed out by Figure 8(c). This is related to the relatively high bandwidth (compared to their) they can sometimes obtain: roughly speaking, they can obtain half the time four times their upload bandwidth.

As a consequence of this efficiency repartition, it is tempting for an average peer to tweak its number of connection in order to increase the efficiency of its connections. For instance, suppressing one connection can improve the probability of collaborating with higher peers. However, this leads to a Nash equilibrium where all peers have just one TFT slot. This is unacceptable in term of connectivity, but rational peers trying to maximize their benefit cannot be avoided. This is an explanation for the 4 slots (3 TFT and one generous slot) settings: obedient average peers that uses the default settings must have at least 4 in order to ensure connectivity in the TFT collaboration graph. On the other hand, the more slots they have, the farther they are from the Nash equilibrium that rational peers will try to follow. Hence 4 seems to be the best trade-off.

7 Conclusion

In this paper, we identified the stable matching theory as a natural candidate to model peer-to-peer networks where peers choose their collaborators. Furthermore, we applied elements of this theory to a specific case: b-matching with global rankings. Whereas there has been a lot of work in analyzing incentive to collaborate in some specific application from an economical point of view, this is the first attempt to analyze the behavior of a class of applications using graph theory.

The main conclusion of this study is that matching theory gave insights on the behavior of a P2P systems class, namely the global ranking class. In both cases of complete and random acceptance graphs, we studied clustering and stratification issues. On most cases, clustering may be prevented using b-matching with enough connections and some standard deviation. But stratification is an intrinsic property of such networks. It seems impossible to overcome it as long as each peer follows the *try-to-collaborate-with-the-best* rule. Interestingly, for random overlay graphs, the crucial parameter is d, the average number of acceptable peers, which makes stratification a flawlessly scalable phenomenon.

As a first application, our results provide some new insights on BitTorrent parameters. They show that best peers have to set up a large number of connections in order to avoid bad download/upload ratio. The *by default* number of collaboration (4) is justified. It allows, to a certain extend, to maintain connectivity in the TFT exchanges and to protect peers using default settings (obedient peers) from peers with optimized settings (rational peers).

When considering the stable properties which emerge, it also become clear that different class of utility functions leads to very different properties. This can be exploited according to the needs of the targeted application. For example, in a peer-to-peer streaming protocol, the most important feature is a small play out delay but a strong stratification, needed to give peers incentive to collaborate, produce a collaboration graph with large diameter (large play out delay). In many cases, combining different utility function will be necessary. Such a combination can, for instance, be achieved by introducing a second type of collaborations depending on a different global ranking or depending on a symmetric ranking such as latency.

Acknowledgment: *The authors wish to thank James Roberts and Dmitri Lebedev for their helpful comments*

References

[1] http://www.emule-project.net/.

[2] Ashwin R. Bharambe, Cormac Herley, and Benkata N. Padmanabhan. Analyzing and improving a bittorrent networkś performance mechanisms. In *Proceedings of IEEE Infocom*, 2006.

[3] Katarína Cechlárová and Tamás Fleiner. On a generalization of the stable roommates problem. *ACM Trans. Algorithms*, 1(1):143–156, 2005.

[4] Bram Cohen. Incentives build robustness in bittorrent. In *Workshop on Economics of Peer-to-Peer Systems*, 2003.

[5] http://www.edonkey2000.com/index.html.

[6] D. Gale and L.S. Shapley. College admissions and the stability of marriage. *American Mathematical Monthly*, 69:9–15, 1962.

[7] Robert W. Irving, Paul Leather, and Dan Gusfield. An efficient algorithm for the "optimal" stable marriage. *J. ACM*, 34(3):532–543, 1987.

[8] Márk Jelasity, Rachid Guerraoui, and Anne-Marie Kermarrec. The peer sampling service: Experimental evaluation of unstructured gossip-based implementations. In *Middleware*, Toronto, Ontario, Canada, October 2004.

[9] Arnaud Legout, Nikitas Liogkas, Eddie Kohler, and Lixia Zhang. Clustering and sharing incentives in bittorrent systems. 2006.

[10] J. Reynier. Erdös-rényi pairing in matlab. `http://www.di.ens.fr/~jreynier/Recherche/MatlabR2.zip`.

[11] Eytan Ronn. *On the complexity of stable matchings with and without ties*. PhD thesis, Yale University, 1986.

[12] S. Saroiu, P. Gummadi, and S. Gribble. A measurement study of peer-to-peer file sharing systems. In *Proceedings of Multimedia Computing and Networking*, 2002.

[13] Jimmy J. M. Tan. A necessary and sufficient condition for the existence of a complete stable matching. *J. Algorithms*, 12(1):154–178, 1991.

Unité de recherche INRIA Rocquencourt
Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex (France)

Unité de recherche INRIA Futurs : Parc Club Orsay Université - ZAC des Vignes
4, rue Jacques Monod - 91893 ORSAY Cedex (France)
Unité de recherche INRIA Lorraine : LORIA, Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex (France)
Unité de recherche INRIA Rennes : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex (France)
Unité de recherche INRIA Rhône-Alpes : 655, avenue de l'Europe - 38334 Montbonnot Saint-Ismier (France)
Unité de recherche INRIA Sophia Antipolis : 2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex (France)

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, BP 105 - 78153 Le Chesnay Cedex (France)
http://www.inria.fr
ISSN 0249-6399