Can P2P Networks be Super-Scalable?

François Baccelli, Fabien Mathieu, Ilkka Norros, Rémi Varloot

Can P2P Networks be Super-Scalable?

François Baccelli
UT Austin & INRIA – ÉNS
USA

Fabien Mathieu
INRIA – University Paris 7
France

Ilkka Norros
VTT
Finland

Rémi Varloot
INRIA
France

***Abstract*—We propose a new model for peer-to-peer networking which takes the network bottlenecks into account beyond the access. This model can cope with key features of P2P networking like degree or locality constraints together with the fact that distant peers often have a smaller rate than nearby peers.**

Using a network model based on rate functions, we give a closed form expression of peers download performance in the system's fluid limit, as well as approximations for the other cases. Our results show the existence of realistic settings for which the average download time is a decreasing function of the load, a phenomenon that we call super-scalability.

I. Introduction

The Peer-to-Peer (P2P) paradigm has been widely used to quickly deploy low-cost, scalable, decentralized architectures. For instance, the success of BitTorrent [1] has shown that file-sharing can be provided with full scalability. Although many other architectures currently compete with P2P (dedicated Content Distribution Networks, Cloud-based solutions, ...), P2P is still unchallenged with respect to its low-cost and scalability features, and remains a major actor in the field of content distribution.

Today, the main limitation for P2P content distribution is probably the access upload bandwidth, as even high-speed Internet access connections are often asymmetric with a relatively low uplink capacity. Therefore most P2P content distribution performance models assume a relatively low access bandwidth as the main performance bottleneck. However, in a near future the deployment of very high speed access (e.g. FTTH) will challenge the justification of this assumption. This raises the need of new P2P models that describe what happens when the access is not necessarily the main/only bottleneck and that allow one to better understand the fundamental limitations of P2P.

A. Contributions

A new model. The first contribution of the present paper is the model presented in Section III, which features the following two key ingredients: 1) a spatial component thanks to which the topology of the peer locations is used to determine their interactions 2) a networking component allowing one to represent the actual exchange throughput between peers.

A promising form of scalability. In most P2P bandwidth models, the upload/download capacity is the bottleneck determining the exchange throughput obtained by peers [2], [3], [4]. This creates *scalability*, where the download latency remains constant when the system load increases. Our new model exhibits a stronger form of scalability, which we call *super-scalability*, where the service latency actually *decreases* with the system load.

We show in Sections II and IV that super-scalability is a consequence of network dynamics causing the service rate of a typical customer to increase with the load of the system.

Conditions for super-scalability to hold. One may question the realism of such a model, as the underlying network obviously cannot sustain arbitrarily high rates. Section V combines our model with an abstract (physical) network model to determine the conditions for which our model makes sense and super-scalability occurs.

Another natural issue is data availability: bandwidth can be a bottleneck only if peers have something to transmit to each other. We address this issue in Section VI, where we study the impact of data availability on the effective download performance.

The laws of super-scalability. Starting from the basic model studied in Section IV, we build in Section VII a Swiss Army Knife for handling many realistic variants: generic rate functions, auxiliary servers, seeding behavior of users, access bottleneck conditions... The corresponding laws determine optimal tuning of the parameters of the P2P algorithms e.g. peering degree, transport protocol or seeding times.

B. Related Work

Our main scenario is inspired by a BitTorrent-like file-sharing protocol. In BitTorrent [1], a file is segmented into small chunks and each downloader (called *leecher*) exchanges chunks with its neighbors in a peer-to-peer overlay network. A peer may continue to distribute chunks after it has completed its own download (it is then called a *seeder*). Here is a short summary of what is kown on this scenario.

Bandwidth-centered modeling. Some studies have analyzed the effectiveness of P2P file-sharing with a simple dynamic system model of peer arrival, focusing on the performance under the assumption that the access bandwidth is the main bottleneck [2], [3], [4]. While the present paper focuses on a similar bandwidth-centered approach, it introduces a richer family of peer interaction models.

Chunk availability. Another potential bottleneck is chunk availability. The worst possible case is the "missing piece syndrome" [5], where one chunk keeps existing in only a few copies (or none!) and the peer population can grow unboundedly while trying to get that chunk. The syndrome may happen for some scenarios [6], [7], but it can be avoided by using more or less sophisticated download policies, at the cost of somewhat increased download times, see [6], [7], [8], [9], [10]. Also note that [11] proposed an elegantly abstracted stochastic chunk-level model of uncoordinated file-sharing. The results in [11] indicate that if the system has high input rate and starts with a large and sufficiently balanced population

of chunks, it may perform for a long time without missing chunk even if there is no seeder.

In this paper, we assume that missing chunk issues are avoided by some mechanism (like getting the *locally rarest* chunk with high priority), so the impact of chunk on performance is reasonable. Nevertheless, we estimate this impact through a very simple chunk-level modeling, inspired by the ones proposed in [3] and [11].

Spatially-dependent rate. While a large number of studies consider the case of heterogeneous rates, to the best of our knowledge, none considers a system where the transfer speeds depend on pair-wise distances but not on the nodes as such. There are some earlier papers considering P2P systems in a spatial framework (for instance, [12]), but they do not assume that distance has some effect on transfer speed. Our paper seems to be the first where a peer's downloading rate is a function of its distances to other peers.

II. Super–scalability Toy Example

Before getting into the core of the paper, consider a system in steady state where peers arrive with some arrival intensity λ, download some file of size F and leave the system as soon as their own download is completed. We neglect here geometry as well as chunk availability issues. By the latter we mean that a peer has always a chunk to provide for another, unfinished peer.

Suppose that the access upload bandwidth is the main bottleneck. If U is the typical upload bandwidth of a peer, then it makes sense to assume that U is also the typical download throughput experienced by each peer. In particular, in the steady state (if any), the mean latency W and the average number of peers N should be such that

$$W = \frac{F}{U} \text{ and } N = \lambda W = \frac{\lambda F}{U} \text{ (Little's Law).} \quad (1)$$

Although very simple, (1) contains a core property of standard P2P systems: the mean latency is independent of the arrival rate. This is the *scalability* property, one of the main motivations for using P2P.

Now, imagine a complete shift of the bottleneck paradigm. Let the main resource bottleneck be the (logical, directed) links between nodes instead of the nodes themselves. We should then consider the typical bandwidth U *from one peer to another* as the key limitation. If each peer is connected to every other one (the interaction graph is complete at any time), then Equation (1) should be replaced by $W = \frac{F}{(N-1)U}$ and $N = \lambda W$, which leads to

$$N = \sqrt{\frac{\lambda F}{U} + \frac{1}{4}} + \frac{1}{2} \text{ and } W = \sqrt{\frac{F}{\lambda U} + \left(\frac{1}{2\lambda}\right)^2} + \frac{1}{2\lambda}.$$

For $\frac{\lambda F}{U} \gg 1$, this can be approximated by

$$N \approx \sqrt{\frac{\lambda F}{U}} \text{ and } W \approx \sqrt{\frac{F}{\lambda U}}. \quad (2)$$

Now, the service time is inversely proportional to the square root of the arrival intensity: this is *super-scalability*.

Remark 1: In fact, the real solution is a little bit more complex than that due to size fluctuations that have not been taken into account here. A more rigorous description of the toy model is available in [13].

TABLE I. Notation for the Basic Model

Name	Description	Units
λ	Leecher arrival rate	$m^{-2} \cdot s^{-1}$
C	Rate parameter	$bits \cdot s^{-1} \cdot m$
F	Mean file size	$bits$
R	Peering range	m
W	Mean latency	s
μ	Mean rate	$bits \cdot s^{-1}$
β	Peer density	m^{-2}

In this toy example, the central reason for super-scalability is rather obvious: the number of edges in a complete graph is of the order of the square of the number of nodes, and so is the overall service capacity.

The main question addressed in the present paper is to better understand the fundamental limitations of P2P systems and in particular to check whether super-scalability can possibly hold in future, network-limited, P2P systems, where the throughput between peers will be determined by transport protocols and network resource limitations rather than the upload capacity alone. This requires the definition of a new model allowing one to capture the toy model idea while taking into account the limitations inherent to P2P overlays as well as network capacity constraints.

III. Network Limited P2P Systems

The aim of this section is to define a basic model that tries to capture super-scalability, spatially dependent rates and P2P constraints. This model will be extended in the last sections of the paper.

Spatial domain. Our peers live in a domain D equipped with a distance d. The meaning of d can be manyfold: physical distance; latency-based pseudo-distance [14]; D can even be some representation of peer categories, the position of a peer representing its own centers of interest. The main point is that we assume that the rate between two peers depends on their distance in D. For simplicity, we focus on a basic model where D is an arbitrarily large torus that approximates the Euclidean plane $\mathbb{R}^2$, but there is no basic difficulty in extending this framework to other topologies better suited to model networks, like a hyperbolic space [15]. Distances in D are expressed in meters, regardless of the actual meaning of D.

Arrival rate. We assume that new peers arrive according to a Poisson process with space-time intensity λ (*"Poisson rain"*). The parameter λ, expressed in $m^{-2} \cdot s^{-1}$, describes the birth rate of peers: the number of peer that arrive in a domain of surface A (expressed in m^2) in an interval $[s,t]$ (in seconds) is a Poisson random variable with parameter $\lambda A(t-s)$.

Data rate. For our basic model, we assume that the transfer rate is determined by a congestion mechanism like TCP Reno. On the path between two peers, let ϑ denote the packet loss probability and RTT the round trip time. Then the square root formula [16] stipulates that the rate obtained on this path is $\frac{\xi}{\text{RTT}\sqrt{\vartheta}}$, with $\xi =\sim 1.309$. Assuming the RTT to be proportional to distance r yields a transfer rate of the form

$$f(r) = \frac{C}{r}, \quad (3)$$

where C is a rate parameter expressed in $bits \cdot s^{-1} \cdot m$.

We assume that the rates are additive, so that the total download rate of a peer x is

$$\mu(x) = \sum_{y \in N(x)} f(d(x,y)), \quad (4)$$

where $N(x)$ is the set of neighbors of x (in the overlay) and $d(x,y)$ the distance between x and y.

We consider symmetric connections, because: the data rate function is symmetric; chunk availability may be neglected for proper parameters (see Section VI); some tit-for-tat mechanisms may be at play to enforce some kind of reciprocity between peers. By symmetry, $\mu(x)$ is also the upload rate of a peer at x. In order for the access not to be a further limitation, the access capacity of a peer at x should exceed $\mu(x)$. This is our default assumption here (access as a possible bottleneck is considered in Section VII).

The choice of a rate function given by (3) is mainly for giving explicit results based on a simple distance-varying rate. Our results indeed apply for a wide range of rate functions (cf. Section VII-A).

Data size. Each peer p wants to get an amount $F_p > 0$ of data. In the basic BitTorrent example where every peer wants to get the same file, F_p would most naturally be modeled by a constant F (the size of the file). For the sake of mathematical tractability, in the analytical models, we follow the approach used by [3] and assume that the F_p's are independent and identically distributed random variables, with finite expectation $F = \mathbb{E}(F_p)$.

Unaltruism. When a peer has finished its download, it leaves the system immediately (instead of becoming a seeder).

Connectivity limitation. The toy example assumes full mesh connectivity between peers, which is not a reasonable assumption. In practice, peers usually limit their neighborhood by using some *overlay graph*. There are many ways to build an overlay, for instance by selecting only peers with sufficient qualities and/or by limiting their total number of neighbors. In the basic model, we propose to define connectivity by a *range* R: if Φ_t is the set of peers present at time t, then $N_t(x) = \{y \in \Phi_t, y \neq x, \text{s.t. } d(x,y) \leq R\}$. The range can for instance originate from an ALTO-like connection management that prevents peers too far from one another to connect [17]. This constraint is even more meaningful in a wireless context, as it can represent the transmission range.

Other connectivity rules could be enforced, for instance random connectivity, but if the rate function decreases with the distance, it is only natural to enforce proximity in the overlay graph. Later in the paper (Section VII), we propose another proximity-based variant where a constant number of closest peers is selected.

Chunks. In order to focus on bandwidth aspects, the basic model follows the approach proposed by [3]: we assume that the effect of chunk (un)availability between peers is that the download effectiveness is affected by some factor $\eta \leq 1$. In the following, we omit η by assuming that file sizes are virtually scaled by a factor $\frac{1}{\eta}$. The actual value of η will be investigated in Section VI.

IV. Study of the Basic Model

In this section, we give some theoretical results for the basic model when D is a subdomain of the Euclidean plane (or a two dimensional torus). We only give here the key ideas that explain the results. Detailed proofs are available in [13].

A. Steady State

The system's dynamics belongs to the class of spatial birth and death processes [18]. The births are the peer arrivals described above. The death rate of a peer at x is $\mu(x)/F$ with $\mu(x)$ given by formula (4). The first result is about the stability of the system:

Proposition 1: If the domain D in which the peers live is compact, then the spatial birth and death process (i.e. the positions of peers present at time t) forms a Markov process which is ergodic for any birth rate $\lambda > 0$.

The proof of Proposition 1 is based on a domination argument. The claim also holds in $\mathbb{R}^2$ but requires a more sophisticated proof that will appear in a forthcoming paper.

According to Proposition 1, the model admits a steady state regime where the peers (in the basic model all leechers) form a stationary and ergodic point process in D [19].

We denote by β_o the density of the peer (leecher) point process, by μ_o the mean rate of a typical peer, by W_o the mean latency of a typical peer, and by N_o the mean number of peers in a ball of radius R around a typical peer, all in the steady state regime of the P2P dynamics.

In the following, we will also consider several approximations of the main model:

- a *fluid regime/limit*, where the corresponding quantities will be denoted by a subscript f (e.g. β_f);
- a heuristic description with a hat notation (e.g. $\hat{\beta}_0$)

In any of these regimes, Little's law tells us that the average density verifies $\beta = \lambda W$.

B. Fluid Limit

The fluid limit consists in assuming that, in the steady state regime, peers are distributed according to an homogeneous Poisson point process in D such that the mean number of neighbors of any peer is large. In particular, in the fluid limit, the presence of a single peer at a given point does not impact the distribution of the other peers.

From Campbell's formula [19], the mean total rate of a typical location of space (or of a newcomer peer) is then

$$\mu_f = \beta_f 2\pi \int_{r=0}^{R} (C/r) r dr = \beta_f 2\pi C R. \quad (5)$$

Now, the fluid limit assumes that a peer sees μ_f during its whole lifetime. We get that the mean latency of a peer is

$$W_f = \frac{F}{\mu_f}. \quad (6)$$

Using Little's law, one gets

$$\beta_f \mu_f = \lambda F. \quad (7)$$

From (5), (6) and (7), we have

$$\beta_f = \sqrt{\frac{\lambda F}{2\pi C R}}, \ \mu_f = \sqrt{\lambda F 2\pi C R}, \ W_f = \sqrt{\frac{F}{\lambda 2\pi C R}}. \quad (8)$$

As we see in the expression for the mean latency in (8), the fluid limit exhibits the same super-scalability as the toy

example: in spite of the fact that the interactions are limited in range and depend on the distance, the mean latency decreases in $\frac{1}{\sqrt{\lambda}}$ when λ tends to infinity and everything else is fixed.

Note that in the fluid limit, the mean number of peers in a ball of radius R around a typical peer is

$$N_f = \pi R^2 \beta_f = \sqrt{\frac{\pi}{2}} \sqrt{\frac{\lambda F R^3}{C}}. \quad (9)$$

C. Dimensional Analysis

At this point of the paper, the fluid limit is a thought experiment, not necessarily related to the actual model. Dimensional analysis [20] helps to connect the two.

In the basic model, the system has 4 parameters (the range R, the file size F, the peer arrival rate λ and the rate parameter C) expressed in 3 basic physical units (meters, bits, seconds). The π-theorem [20] allows us to strip the problem from all its parameters but one. The idea is that the behavior of a system is not affected by the physical units used to measure it. By using proper unit changes [13], the system can be described by just one dimensionless parameter

$$\rho = \frac{\lambda F R^3}{C}. \quad (10)$$

The π-theorem leaves some freedom in the choice of the parameter. By noticing that $N_f = \sqrt{\frac{\pi}{2}}\sqrt{\rho}$, we can use N_f, which has a physical interpretation (the number of neighbors predicted by the fluid limit), instead of ρ.

The π-theorem tells us that all systems that share the same parameter N_f are similar. Now consider the union of two independent systems that use the same parameters (λ, F, C, R): the real model, with latency W_o, and the fluid model, with latency W_f. The ratio $\frac{W_o}{W_f}$ is a dimensionless property of the overall system, therefore it is a function of N_f only. In other words, there exists a dimensionless function $M(N_f)$ such that:

$$W_o = M(N_f) W_f. \quad (11)$$

From Little's law, we also deduce the density:

$$\beta_o = \beta_f M(N_f). \quad (12)$$

Note that the dimensional reasoning made on the basic model can be extended to other models, for instance with different rate functions or connectivity rules. Equation (12) will remain true, although the shape of M may change; in particular, if the system is described by more than 4 parameters, M may depend on more than one variable.

To summarize, although the system in the basic model may be subject to complex interactions and is defined by four independent parameters, dimensional analysis allows one to express its general behavior through a one-parameter function M (unknown at this point), which expresses how far the actual system is from its fluid limit.

D. Fluid as a Bound

We now give a better understanding of the behavior of the real system through the following theorems.

Theorem 1 (Fluid as a bound): $M \geq 1$. In other words, the fluid regime is actually a lower bound for the mean latency and the peer density.

The proof comes from a stochastic intensity argument. This property stems from the fact that as a peer uploads content to its neighbors, it makes them leave the system faster than if it did not upload anything. This is called a *repulsion* effect. As a result, the mean download rate experienced by a typical peer (Palm distribution) is less than the mean download rate that would experience a virtual, non uploading, peer located at a typical location of D. Details can be found in [13].

Theorem 2 (Fluid as a limit): When N_f goes to infinity, M goes to 1, and the law of a typical peer latency converges weakly to an exponential random variable with parameter $1/W_f$.

Theorem 2 says that the fluid bound is tight: when the number of neighbors predicted by the fluid limit tends towards infinity, the system behaves like its fluid limit.

The idea of the proof is that, when N_f tends to infinity: (i) the traffic is high enough for the impact of one given peer, and thus the repulsion effect, to be neglected; (ii) the peers stay long enough to make the fluctuations slow and weak. The fact that the rate at any point is constant in the limit implies that the latency is exponential in the limit.

E. Heuristic

For arbitrary values of N_f, we propose to approximate M by $\hat{M}$, the unique solution in $[1, \infty)$ of

$$\hat{M}^2 \left(1 - \frac{\hat{M}}{2N_f} \ln\left(1 + \frac{2N_f}{\hat{M}}\right)\right) = 1. \quad (13)$$

In order to derive (13), we use a heuristic factorization of the factorial moment measure of order 3 of the stationary peer point process (see [19] for the definition of these measures) which is described in [13]. Informally, the method consists in computing an approximation $\hat{u}_o$ of the average rate of a peer assuming that: (i) a neighbor at distance r from that peer "sees" a rate $\hat{u}_o + \frac{C}{r}$; (ii) in return, the peer "sees" at distance r a density of neighbors $\frac{\lambda F}{\hat{u}_o + \frac{C}{r}}$ (using (7)).

This heuristic is in line with Theorems 1 and 2.

Remark 2: When N_f goes to 0, the system admits another limit, called hard-core*, which was not presented here due to its lack of interest for real P2P systems. Nevertheless, the heuristic is in line with the hard-core limit too, which predicts that M behaves like $\frac{1}{N_f}$ when N_f goes to 0 [13].*

F. Validation

We validated and substantiated our results by means of simulations of our model. We used a discrete time simulator to evaluate the basic model for several values of N_f (see [13] for details). Key results are displayed in Figure 1, which allows us to check almost all results of this section in one look:

- $M = 1$ is a lower bound of the actual system (Theorem 1);
- as N_f goes to ∞, the bound becomes tight (Theorem 2);
- the heuristic (13) gives a good approximation of M;
- as N_f goes to 0, the system behavior converges towards the hard-core limit $M = \frac{1}{N_f}$ (cf. Remark 2).

We also checked that for N_f big enough, it is quite difficult to distinguish the system from a spatial birth and death process with birth parameter λ and death parameter $1/W_f$, namely a Poisson point process of intensity β_f (cf. [13]).

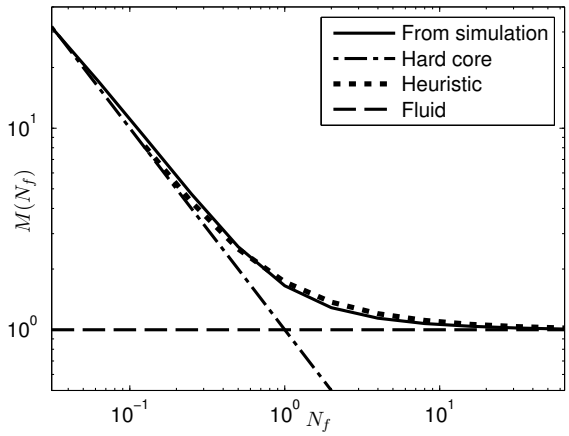

Fig. 1. $M(N_f)$ in the basic model.

V. Network Capacity Constraints

Super-scalability naturally rises the question of the burden on the underlying network. The aim of this section is to determine the capacity required for the network elements in order to achieve the super-scalable regime identified above.

So far, the only assumptions on the network were that 1) the access is not the (only) bottleneck; 2) the network is a bottleneck, resulting into a transfer rate between peers that depends on their distance.

This section introduces an abstract network model on which the P2P traffic will be mapped through some natural shortest path routing mechanism. We determine the mean *flow* that traverses a typical network element. This flow of course depends on the protocols used in the network which in turn determine the bit rate function.

For simplicity, we consider the fluid limit of the system.

A. Network Model

We consider an underlying network made of routers and links between them where

- routers form a realization of a spatial Poisson point process of intensity θ;
- links are the Delaunay edges (see e.g. [21], Chapt. 4) on this point process;
- the capacity of a link is E;
- each peer is directly connected to the closest router and the path between two routers is a minimal path (with respect to hop count) on the Delaunay graph.

In this case, the number of links between two peers is asymptotically proportional to the distance between them [21].

Consider a straight line of the plane of length l. The average number of links that go through the line is $2l\sqrt{\theta}$, so the maximal traffic that can cross the line is $2El\sqrt{\theta}$. In other words, $\Xi := 2\sqrt{\theta}E$ is a parameter that describes the capacity of the network, expressed in $bits \cdot s^{-1} \cdot m^{-1}$.

B. Flow Equations

Let $\Psi(\varepsilon)$ denote the mean value of the P2P traffic that goes through a segment S of length ε in the fluid regime. By isotropy, we can focus on $S = [(0, -\frac{\varepsilon}{2}), (0, \frac{\varepsilon}{2})]$.

A simple stochastic geometry argument shows that

$$\Psi = \Psi(1) = 4\beta_f^2 \int_0^R r^2 f(r) dr \tag{14}$$

(see [13]). Using the fluid expression of the density

$$\beta_f = \sqrt{\frac{\lambda F}{2\pi \int_0^R r f(r) dr}},$$

we get the key relation

$$\Psi = \Psi(1) = \frac{2}{\pi} \lambda F \frac{\int_0^R r^2 f(r) dr}{\int_0^R r f(r) dr}. \tag{15}$$

Equation (15) holds for an arbitrary rate function f. For $f(r) = \frac{C}{r}$, we get

$$\Psi = 2C\beta^2 \varepsilon R^2 = \frac{1}{\pi} \lambda F R. \tag{16}$$

C. Feasibility Condition

Now, in order to simplify the evaluation of the P2P load on the underlying network, we assume that (a) θ is large enough so that the hop-count between two peers can be seen as proportional to their distance and the flow between them as a straight line; (b) Any rate smaller than Ξl can be transported through a segment of length l. Under these assumptions, the condition for the network to sustain the rate generated by our model is

$$\Psi < \Xi. \tag{17}$$

Note that the flow Ψ in (16) does not depend on C, so that condition (17) does not either. This surprising result means that in the fluid limit, we can arbitrarily scale the individual rate of connections (thus decreasing the latency) without changing the burden on the underlying network. Of course, there is a flaw in that reasoning: increasing C eventually impairs the validity of the fluid limit. As C increases, N_f gets smaller so we tend to leave the fluid limit and the approximations we used do not apply anymore [13].

VI. Adding Chunks to the Model

This section contains a mathematical model and a simulation study allowing one to quantify the impact of chunk availability. An important result is that when both the number of chunks and the parameter N_f (introduced in Section IV) are large, then the systems behaves as the chunkless fluid model of Section IV.

A. Chunk Modeling

We assume now that the file has a constant size F and is divided into K chunks of equal length. At any time, a peer is characterized by its *collection*, which is the subset of chunks it fully possesses. With respect to dimensional analysis, the system is now described by two parameters: N_f and K.

For simplicity, we focus on the steady state taken in its fluid limit with respect to the peers, and we assume that the chunk scheduling policy is based on the following principles:

- *rarest chunk first*: when a peer can choose between chunks to download, it selects the one with fewest copies in its neighborhood; as in [3], we assume that this prevents the missing chunk syndrome and ensures that a peer with k chunks has a collection of chunks which is independent of that of the

other peers and uniform on the subsets of cardinality k of the set $\{1, \ldots, K\}$;

- *random peer order*: when it can download a given chunk from many neighbors within its range, a peer chooses one at random (the scheduling is not *network-aware*).

There are two main ways to manage the download of simultaneous chunks: in the *one-to-one* model, a peer gets one chunk from a single neighbor, while in the *many-to-one* model, it can aggregate the resources of all neighbors that possess that chunk. The many-to-one approach gives better theoretical performance, as we will see below, but it requires a tight synchronization between peers that collaborate for a chunk, and thus may require an additional overhead in practice.

B. Performance Study

An exhaustive study would require to consider the $2^F - 1$ possible collections (although seeders are initially needed to bootstrap the system, we still consider a steady state with no seeder, so there is no full collection). With the proposed assumptions, the impact of chunks mainly depends on the number of chunks already possessed by the peers. We say that a peer belongs to class k, for $0 \leq k \leq K-1$, if it possesses exactly k complete chunks. The following theorem gives the performance of each class in the fluid regime (by fluid regime, we mean i) a chunk regime where the independence and uniformity assumptions described above on the distribution of the chunks hold and ii) a peer regime where the Poisson assumptions described in the preceding section hold).

Theorem 3: In the fluid limit, the mean total download rate of a peer of class k, $0 \leq k \leq K$, is

$$\mu_k = \eta_k \mu_f, \tag{18}$$

where μ_f is given by (8). Equation (22) gives the η_k's for the many-to-one scheduling while (24) gives a lower bound for the one-to-one case.

Proof: In view of our assumptions on the scheduling and on the distribution of peers, the average rate of a given transfer is just the average over the range, that is $\frac{1}{\pi R^2} \int_0^R 2\pi r (C/r) dr = \frac{2C}{R}$.

Now, we consider a peer p of class k with a neighbor q of class j. In view of our assumptions on the distribution of chunks, the probability that q has at least one chunk that p wants, which coincides with the probability that the set of chunks of q is not included in that of p, is

$$z(k,j) = 1 - \binom{k}{j} / \binom{K}{j}, \tag{19}$$

with the convention that $\binom{k}{j} = 0$ for $j > k$. Thus, if β_j denotes the density of class j, the number of neighbors from whom a given peer of class k may download one chunk is

$$N_c = \pi R^2 \sum_{j=0}^{K-1} \beta_j z(k,j). \tag{20}$$

In the many-to-one model, we deduce that the average download is

$$\mu_k = \frac{2C}{R} \pi R^2 \sum_{j=0}^{K-1} \beta_j z(k,j). \tag{21}$$

We notice then that for class k, (7) becomes $\beta_k = \frac{\lambda F}{K \mu_k}$. To conclude, we define $\eta_k := \frac{\mu_k}{\mu_f}$, where μ_f is given by (8). If we replace β_k by $\frac{\lambda F}{K \mu_k}$ in (21) and use the relationships from (8) and (9), we get

$$\eta_k = \frac{1}{K} \sum_{j=0}^{K-1} \frac{z(k,j)}{\eta_j}. \tag{22}$$

In the one-to-one model, a peer cannot download a chunk from more than one peer. In the worst case where each of the N_c peers has at most one of the desired chunks, the probability that p can download any given desired chunk is $1 - (1 - \frac{1}{K-k})^{N_c}$, so that the average number of chunks downloaded is

$$(K-k) \left(1 - (1 - \frac{1}{K-k})^{N_c} \right). \tag{23}$$

Adapting (21), using the same variable changes as for the many-to-one case, and using N_f as a lower bound for N_c, one gets:

$$\eta_k \geq \frac{K-k}{N_f} \left(1 - (1 - \frac{1}{K-k})^{N_f} \right). \tag{24}$$

■

Equation (22) is easily solved using fixed-point iterations. Notice that the computation depends solely on K in the many-to-one model and on K and N_f in the one-to-one model. If η denotes the harmonic mean of the η_k's, we verify that the overall latency W is $\frac{W_f}{\eta}$. Therefore, as for the model proposed in [3], η can be used to scale the results of the basic model and ignore the underlying, possibly complex, chunk exchange mechanisms.

Remark 3: In the basic model we had $W = M(N_f) W_f$, so we can interpret $\frac{1}{\eta}$ as $M(N_f, K)$ in the case $N_f \gg 1$.

We now study the behavior of η in the fluid limit.

Theorem 4: In the many-to-one model, and in the one-to-one if N_f is large enough yet fixed, we have

$$\eta \xrightarrow[K \to \infty]{} 1. \tag{25}$$

Sketch of Proof: For the many-to-one model, we use a scaling technique that consists in letting K go to infinity so as to make the η_k converge toward a continuous function in $[0, 1)$. The basic ingredient is the fact that the function z defined in (19) converges pointwise to 1 under this scaling. The scaling of (22) is

$$\eta(x) = \int_0^1 \frac{1}{\eta(y)} dy. \tag{26}$$

It is not difficult to show that $\eta = 1$ is the unique positive solution solution of this functional equation, which proves (25) for the many-to-one case.

In the one-to-one model, (25) is straightforward when noticing that η is always smaller than or equal to 1 (the overall download capacity is lowered because of availability issues). The limit of (24) when K tends to ∞ allows one to conclude. ■

The fact that a peer cannot upload a given chunk from more than one peer badly impacts the performance of the one-to-one model, compared to many-to-one. This is especially true at the end of the download, when a peer may have more useful neighbors than remaining chunks. This fact was empirically observed by Bram Cohen in his original BitTorrent design, where he proposed to use one-to-one (which is easier to maintain) most of the time except for the very few last chunks, where peers switch to many-to-one (endgame behavior [1]).

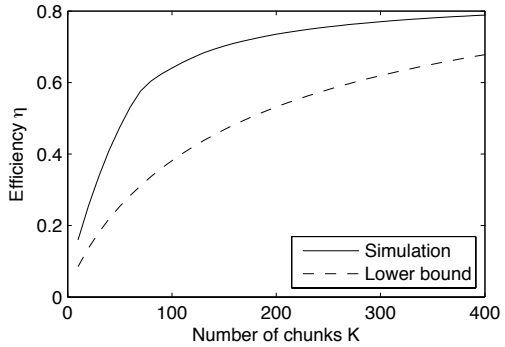

Fig. 2. Efficiency η as a function of K ($N_f = 40$).

C. Validation

We simulate the system with chunks in order to substantiate our claims, using a simple rarest first chunk selection and random peer selection like the one proposed. Synchronization is one-to-one.

First, we validate the assumption on the distribution of chunks by checking the impact of the presence of a chunk at some peer on the presence of this chunk at the neighboring peers. For instance, for $N_f = 40$, $K = 200$, we verified that a peer sees in average 29.22 copies of a chunk it possesses (itself not included), and 29.10 copies of a chunk it misses. This and more detailed correlation analysis (that cannot be included here due to space limitation) are quite conclusive.

We launched many trials to verify our results. Figure 2 displays the value of η for several values of K. One verifies that the system has a better performance than the proposed lower bound, and the right behavior when K grows.

D. Conclusion on Chunks

We showed (through analysis and simulation) that in the fluid limit ($N_f \gg 1$), when $K \gg 1$, the system with chunks behaves like the fluid chunkless model of Section IV with an appropriate efficiency parameter η, which we described.

The parameter η can be close to 1 if K is large enough, with N_f being fixed in the one-to-one model. In this last case, super-scalability could be impacted: as λ increases, so does N_f and if K is fixed, the lower bound converges to 0 (simulations confirm that this is also the case for η). The possible workarounds for this issue are: to use many-to-one, or equivalently one-to-one with endgame, to get rid of the last chunks bottleneck; to limit the number of neighbors in order to keep N_f bounded (this will be detailed in Section VII-F).

VII. Extensions of the Basic Model

The aim of this section is to show that our analysis can be extended in several ways and take important practical phenomena into account. Unless otherwise stated, we will place ourselves in the fluid regime, but the dimensional analysis approach can be used with all extensions to relate the fluid limit to the real system through some function M. As we have seen when introducing the chunks, if an extension introduces new parameters, M can be a function of several dimensionless variables (replacing N_f).

For sake of clarity, the proposed extensions are presented separately, but interleaving extensions is straightforward in the fluid limit. Outside the fluid limit, the complexity of mixed extensions will mainly depend on the complexity of the corresponding M function.

TABLE II. Some rate functions with explicit strength γ

$f(r)$	Interpretation	$\gamma = 2\pi \int_0^R rf(r)dr$
$\frac{C}{r}$	TCP-like	$2\pi CR$
U	UDP-like (constant)	πUR^2
$\frac{C}{r} \wedge U$	TCP with per-flow limitation	$\pi\left(2CR - \frac{C^2}{U}\right)$ [a]
$\frac{C}{r+q}$	TCP with offset	$2\pi C\left(R - q\ln(1+\frac{R}{q})\right)$
$\frac{C}{r} - o$	TCP with overhead	$\pi R\,(2C - oR)$ [b]
$\frac{1}{2}\ln\left(1+\frac{C}{r^\alpha}\right)$	SNR Wireless	$\frac{\pi^2 C^{\frac{2}{\alpha}}}{2\sin\left(\frac{2\pi}{\alpha}\right)}$ for $R=\infty$ [c]

[a] For $C \le UR$; $C \ge UR$ is the UDP-like case.
[b] For $\frac{C}{R} \ge o$; otherwise replace R by $\frac{C}{o}$.
[c] There is no closed form for$R < \infty$ in most cases. However, for $\alpha = 4$, we have $\gamma = \pi\left(R^2\log(1+\frac{C}{R^4}) + \sqrt{C}\arctan(\frac{R^2}{\sqrt{C}})\right)$.

A. More General Rate Functions

While we focused for the basic model on the rate function (3), all our results can easily be generalized to any rate function f such that $\int_{r=0}^R rf(r)dr < \infty$.

For a rate function f, the fluid rate Equation (5) becomes

$$\mu_f = \beta_f\gamma, \text{ with } \gamma = 2\pi\int_{r=0}^R rf(r)dr. \tag{27}$$

The characteristic γ, which is expressed in $bits \cdot s^{-1} \cdot m^2$, is the sum of f over its range, so we call it the *strength* of f. Once γ is known, we can generalize (8) as

$$\beta_f = \sqrt{\frac{\lambda F}{\gamma}},\ \mu_f = \sqrt{\lambda F\gamma},\ W_f = \sqrt{\frac{F}{\lambda\gamma}}. \tag{28}$$

We observe that the scaling in $\frac{1}{\sqrt{\lambda}}$ still holds. For the rest of the paper, we use directly the strength γ instead of (3).

Table II gives the strength of the following rate functions:

- The TCP-like example of the basic model;
- Constant rate function, where each flow has a bandwidth U. This corresponds for instance to the case where the transport protocol is UDP and bandwidth is limited by the application;
- Mix of the above, where the rate is TCP-like with an upper bound set by the application;
- TCP-like with some additive offset q that accounts for the mean delay in the two access networks;
- Capacity of a wireless AWGN channel.

In most cases, the heuristic approximation $\hat{M}$ can be adapted to f. For instance, a constant f leads to (cf [13])

$$\hat{M} = \sqrt{1 + \left(\frac{1}{2N_f}\right)^2} + \frac{1}{2N_f}. \tag{29}$$

If $R = \infty$, the system parameter $N_f = \pi R^2\beta_f$ is not properly defined anymore, which impairs a direct introduction of M. If $\int_{r>0} r^2 f(r)dr < \infty$, a simple workaround is to use the following ratio (already considered in (15))

$$\tilde{R} := \frac{\int_{r>0} r^2 f(r)dr}{\int_{r>0} rf(r)dr} \tag{30}$$

instead of R and to extend the dimensional analysis accordingly ($\tilde{R}$ being interpreted as the *typical* range of f). If $\int_{r>0} r^2 f(r)dr = \infty$, then according to (14) the traffic load intensity is infinite, so the rate function is probably ill-defined with respect to the underlying, capacity-limited, network.

B. Permanent Servers

The system may benefit from servers, or eternal seeders[1]. For instance they can be introduced to: (i) solve the issue of chunk availability by being able to provide any asked chunk; (ii) allow to consider hybrid systems that combine classical server solutions and a P2P approach; (iii) avoid the fact that in our model, the latency goes to ∞ when λ goes to 0 (non-popular content syndrome).

We focus on the basic model.

The servers are characterized by their density of bitrate U_C, expressed in $bit \cdot s^{-1} \cdot m^{-2}$, so that if β_f is the peer density, a typical peer gets $\frac{U_C}{\beta_f}$ from the servers.

To describe the system, we need another dimensionless parameter in addition to N_f. We conveniently choose $\chi := \frac{U_C}{\lambda F}$, which expresses the ratio between the density of rate needed by the system and the density of rate provided by the servers. If $\chi \geq 1$, then the permanent rate from servers is sufficient to serve the peers, otherwise P2P transfer is needed for stability.

Let us focus on the two limiting cases: the system is mainly client/server ($\chi \gg 1$), or the system is mainly P2P with a small server-assistance ($\chi \ll 1$). The case $\chi \ll 1$ can be seen as a scenario where servers are here mainly for insuring chunk availability.

If $\chi \gg 1$, then almost all resources come from the servers. This implies that the point process is hard–core (a peer sees almost no neighbor in its range while it is a leecher, otherwise the P2P traffic would not be negligible), so a peer can collect all the available bandwidth in its range. We deduce the average latency:

$$W_C \approx \frac{F}{\pi R^2 U_C}. \tag{31}$$

For $\chi \ll 1$, in the fluid limit ($N_f \gg 1$), we can adapt (5), which gives

$$\mu_{f,C} = \beta_{f,C}\gamma + \frac{U_C}{\beta_{f,C}}, \tag{32}$$

from which we deduce

$$W_{f,C} = \sqrt{\frac{F - \frac{U_C}{\lambda}}{\lambda\gamma}} = W_f\sqrt{1-\chi} \approx W_f. \tag{33}$$

C. Abandonment

Here we consider the case where all leechers have some abandonment rate. Let a denote this rate. In the stationary state, we have $\lambda = (\frac{\mu_f}{F} + a)\beta_f$. From (27), we deduce $\mu_f^2 + \mu_f aF = \lambda F\gamma$. The positive solution of this equation is

$$\mu_f = \sqrt{\lambda F\gamma + \left(\frac{aF}{2}\right)^2} - \frac{aF}{2}. \tag{34}$$

The analysis can hence be extended without difficulties. For instance, the abandonment ratio is given by $\frac{aF}{\mu_f + aF}$.

[1]This is distinct from the case where leechers can seed for some time after they complete their download, which is addressed in VII-E

D. Per Peer Rate Limitation

Due to the asymmetric nature of certain access networks (e.g. ADSL), the uplink rate is often the most important access rate limitation. Let U denote (here) the average upload capacity of a peer; then the average rate in the fluid limit should be such that

$$\mu_f = \sqrt{\lambda F\gamma} \leq U. \tag{35}$$

If $\gamma = 2\pi RC$ (basic model), a dimensioning rule could be to choose $R = \frac{U^2}{\lambda F 2\pi C}$ so that all available capacity is used.

E. Leechers and Seeders

When a leecher has obtained all its chunks, it can become a seeder and remains such for a duration T_S. In this setting, there is a density of seeders λT_S in the stationary regime.

In the fluid limit with seeders, (27) becomes

$$\mu_{f,S} = (\beta_{f,S} + \lambda T_S)\gamma. \tag{36}$$

Using (7) and $F = W_{f,S}\mu_{f,S}$, we get

$$W_{f,S}^2 + W_{f,S}T_S = W_f^2. \tag{37}$$

The positive solution of this equation is

$$W_{f,S} = \sqrt{W_f^2 + \left(\frac{T_S}{2}\right)^2} - \frac{T_S}{2}. \tag{38}$$

In particular, we have $W_{f,S} \approx W_f$ for $T_S \ll W_f$ and $W_{f,S} \approx \frac{W_f^2}{T_S}$ for $T_s \gg W_f$.

Remark 4: Seeders can also greatly improve the performance in the case where N_f is small, by ensuring that a leecher can find peers in its range with high probability (cf [13] for more details).

F. Limited Degree

In the basic model, we limit connectivity by range for mathematical tractability, but in practice, most P2P systems use a limitation based on the number of connections per peer.

However, degree limited connectivity can be linked to our model. Consider that a ALTO-like mechanism allows each peer to connect to its L *nearest peers*. If L is high enough, it will be identified to N_f and the behavior will be fluid. The degree connectivity can then be approximated by a range connectivity such that L, R and β verify

$$\pi R^2 \beta = L. \tag{39}$$

Using (7) and (27), we get an equation that β must verify:

$$\beta^2\gamma(\beta) = \lambda F, \tag{40}$$

where $\gamma(\beta)$ is the strength of the rate function f when using $R = \sqrt{\frac{L}{\pi\beta}}$ (see for instance Table II).

Once β is known, we deduce $W = \frac{\beta}{\lambda}$. For instance, using the rate function of the basic model, one gets

$$W = \left(\frac{F}{2C}\right)^{\frac{2}{3}} \left(\frac{1}{\pi\lambda L}\right)^{\frac{1}{3}}. \tag{41}$$

We observe that the super-scalability property still exists (although slightly diminished), despite the fact that each peer has a limited number of neighbors. This is a consequence of having a decreasing f function: as the arrival rate increases, so does the density and thus the rate of individual connections. To compare with, a system with a constant rate function like in the toy example is simply scalable if the degree connectivity is limited (the latency is obviously $W = \frac{F}{LU}$).

Finally, we can propose a fluid model that encompasses both the range and degree models. Consider that there is a function $p(r, \beta)$ that describes the probability that a peer connects to another one given that their distance is r and the density is β.

The equation to solve is still (40), except that we now define

$$\gamma(\beta) = \int_{r>0} 2\pi r f(r) p(r, \beta) dr. \tag{42}$$

Under this formalism, the range model is simply $p(r, \beta) = 1_{r \leq R}$, while the degree limited model corresponds to $p(r, \beta) = 1_{r \leq \sqrt{\frac{L}{\pi \beta}}}$. For these two cases, the function p corresponds to very simple overlays, but it could be used to model more complex structures like random geometric graphs.

VIII. Conclusion

In a P2P system with a rate function f and a range R, the following general law quantifying P2P super-scalability was identified: the stationary latency is of the form

$$W_o = M\left(\sqrt{\frac{\pi^2 R^4 \lambda F}{\gamma}}\right)\sqrt{\frac{F}{\lambda\gamma}}, \tag{43}$$

with $\gamma = 2\pi \int_0^R f(r)dr$ and with $M(x)$ a function which is larger than 1 and tends to 1 when x tends to infinity. In the TCP case, the function $x \to M(x)$ is decreasing and hence reinforces super-scalability.

The conditions for the super-scalability formula (43) to hold were also identified: (1) The number of chunks should be large (so as to be in the fluid regime w.r.t. chunks); (2) The parameter $N_f = \pi R^2 \sqrt{\lambda F/\gamma}$ should be large (so as to be in the fluid regime w.r.t. peers). If (1) or (2) do not hold, then chunk/peer availability issues will dominate and the model breaks down; (3) the network should have the capacity to cope with the P2P traffic, i.e.

$$E\sqrt{\theta} > \frac{2\lambda F}{\gamma} \int_0^R r^2 f(r) dr, \tag{44}$$

where θ is the spatial intensity of routers and E the typical link capacity. Hence the capacity of the network should scale like λ if other parameters are unchanged. If this condition does not hold, the network cannot cope with the traffic and the model breaks down; (4) The access should not be the bottleneck, which translates into the requirement

$$U > \sqrt{\lambda F \gamma}, \tag{45}$$

where U denotes the (total) upload capacity of each peer. In other words, the latter should scale like $\sqrt{\lambda}$. If this is not the case, then classical access bottleneck model should be used.

Acknowlegments: This work has been carried out at LINCS (http://www.lincs.fr) and has been partly funded by the EIT ICT Labs Projects *Fundamentals of Networking* and *Distributed Content Delivery in Wireless Networks*.

References

[1] B. Cohen, "BitTorrent specification," 2006, http://www.bittorrent.org.

[2] G. D. Veciana and X. Yang, "Fairness, incentives and performance in peer-to-peer networks," in *In the Forty-first Annual Allerton Conference on Communication, Control and Computing*, 2003.

[3] D. Qiu and R. Srikant, "Modeling and performance analysis of BitTorrent-like peer-to-peer networks," *ACM SIGCOMM Computer Communication Review*, vol. 34, no. 4, pp. 367–378, 2004.

[4] F. Benbadis, F. Mathieu, N. Hegde, and D. Perino, "Playing with the bandwidth conservation law," in *IEEE P2P*, 2008, pp. 140–149.

[5] F. Mathieu and J. Reynier, "Missing piece issue and upload strategies in flashcrowds and P2P-assisted filesharing." in *AICT/ICIW'06*, 2006.

[6] B. Hajek and J. Zhu, "The missing piece syndrome in peer-to-peer communication," 2010, http://arxiv.org/abs/1002.3493.

[7] J. Zhu and B. Hajek, "Stability of a peer-to-peer communication system," *IEEE Transactions on Information Theory*, vol. 58, no. 7, pp. 4693–4713, 2012.

[8] H. Reittu, "A stable random-contact algorithm for peer-to-peer file sharing," in *IFIP IWSOS*, 2009, pp. 185–192.

[9] I. Norros, H. Reittu, and T. Eirola, "On the stability of two-chunk file-sharing systems," *Queueing Systems*, vol. 67, pp. 183–206, 2011.

[10] B. Oğuz, V. Anantharam, and I. Norros, "Stable, scalable, decentralized P2P file sharing with non-altruistic peers," 2011, arXiv:1107.3166v1.

[11] L. Massoulie and M. Vojnovic, "Coupon replication systems," *IEEE/ACM Trans. Networking*, vol. 16, no. 3, pp. 603–616, 2005.

[12] R. Susitaival, S. Aalto, and J. Virtamo, "Analyzing the dynamics and resource usage of P2P file sharing systems by a spatio-temporal model," in *P2P-HPCS06, in conj. with ICCS*, May. 2006, pp. 420–427.

[13] F. Baccelli, F. Mathieu, and I. Norros, "Spatial Interactions of Peers and Performance of File Sharing Systems," INRIA, Research Report RR-7713, May 2012. [Online]. Available: http://hal.inria.fr/inria-00615523

[14] P. Fraigniaud, E. Lebhar, and L. Viennot, "The inframetric model for the internet," in *IEEE INFOCOM)*, 2008, pp. 1085–93.

[15] M. Boguñá, F. Papadopoulos, and D. Krioukov, "Sustaining the Internet with Hyperbolic Mapping," *Nature Communications*, vol. 1, no. 62, Oct 2010.

[16] T. Ott, J. Kemperman, and M. Mathis, "The stationary behavior of ideal TCP congestion avoidance," *Internetworking: Research and Experience*, vol. 11, pp. 115–156, 1992.

[17] J. Seedorf and E. Burger, "Application-Layer Traffic Optimization (ALTO) Problem Statement," RFC 5693 (Informational), Internet Engineering Task Force, Oct. 2009.

[18] C. Preston, "Spatial birth-and-death processes," *Bull. Inst. Internat. Statist.*, vol. 46, no. 2, pp. 371–391, 405–408, 1975.

[19] D. J. Daley and D. Vere-Jones, *An Introduction to the Theory of Point Processes*. Springer, 1988.

[20] E. Buckingham, "The principle of similitude," *Nature*, vol. 96 (2406), pp. 396–397, 1915.

[21] F. Baccelli and B. Błaszczyszyn, *Stochastic Geometry and Wireless Networks, Volume I and II*, ser. Foundations and Trends in Networking. NoW Publishers, 2009.