

Riikka Susitaival, Samuli Aalto, and Jorma Virtamo. Analyzing the dynamics and resource usage of P2P file sharing by a spatio-temporal model. In Proceedings of the 2006 International Workshop on P2P for High Performance Computational Sciences (P2P-HPCS 2006) in conjunction with the 6th International Conference on Computational Science (ICCS 2006), pages 420-427, May 2006.

© 2006 Springer Science+Business Media

Reprinted with kind permission of Springer Science+Business Media.

# Analyzing the Dynamics and Resource Usage of P2P File Sharing by a Spatio-temporal Model

Riikka Susitaival, Samuli Aalto, and Jorma Virtamo

Helsinki University of Technology,  
P.O. Box 3000, FIN-02015 TKK, Finland  
{riikka.susitaival, samuli.aalto, jorma.virtamo}@tkk.fi

**Abstract.** In this paper we study the population dynamics and resource usage optimization of a P2P file sharing system, where the availability of the requested file is not guaranteed. We study the system first by a deterministic fluid model and then by a more detailed Markov chain analysis that allows estimating the life time of the system. In addition, the underlying topology of the network is modelled by a simple geometry. Using the resulting spatio-temporal model we assess how much the resource usage of the network can be reduced, e.g., by selecting the nearest seed for download instead of a random one.

## 1 Introduction

Peer-to-peer (P2P) applications, such as file sharing, have become a significant area of Internet communication in recent years. Older examples of these applications are Gnutella, Napster and Kazaa, whereas BitTorrent is currently the most popular system. It has been widely reported that P2P related traffic forms a major part of the total traffic in the Internet. From an operator's point of view it is important that the traffic load produced by P2P applications does not encumber the underlying network too heavily. Efficient usage of the network resources would also improve the service of the individual peers by shortening average latencies.

We concentrate on BitTorrent-like P2P protocol because of its popularity but the results are applicable to other protocols as well. The idea of BitTorrent is to divide the file to be distributed into parts, named *chunks*, so that different parts can be downloaded from several peers simultaneously, where the size of the chunk is typically 256 KB, see for technical details of BitTorrent in [1]. Measurement studies [2], [3], [4], have shown that the evolution of a single file in the system can be divided into three phases. In the first *flash crowd* phase the demand for the newly released file is high. It is followed by a *steady state* and finally, the *end* means the death of the file.

A few papers have analyzed P2P file sharing systems by stochastic models so far. In paper [5], the analysis of BitTorrent-like system is divided into transient and steady state regimes. The service capacity of the transient regime is studied by a branching process and the steady state by a Markov model. Paper [6] studies

the performance of the system by a deterministic fluid model, whereas in paper [7] the network level latencies are modeled by the delay of a single class open queueing network and peer level latencies by the delay of M/G/1/K processor sharing queues. However, these models do not capture all aforementioned phases of the sharing process, namely flash crowd, steady state and especially end phase.

In this paper we study the dynamics of sharing a chunk, that is, a single piece of a file, in a P2P system. First we model the system by a deterministic fluid model and study the dynamics of the average number of downloader and seeds over time. The deterministic fluid models are, however, unable to capture all the details of the chunk sharing process such as possible instability and extinction of the system. For this reason we construct a complete Markov chain model to obtain more information of the life cycle of chunk sharing process.

By providing the downloaders and seeds with location information we study further how the selection of the peer has an effect on the resource usage in the network. We propose a spatio-temporal model for the P2P system, in which the topology of the Internet is abstracted by a sphere, on which peers are located. Distance metric between two peers in terms of delay or bandwidth is assimilated with their geometrical distance. We consider two different peer selection policies; in the first one a random seed is selected whereas in the second the nearest one is searched. Expected values for the capacity usage for these two peer selection policies are derived and also the dynamics of the system is studied by simulations.

The paper is organized as follows: In section 2 population dynamics of the system is studied by a fluid model. Then a Markov chain model for calculating the time to extinction is constructed in section 3. In section 4 the geometric approach for modeling of chunk sharing is introduced and different peer selection policies are compared. Finally, in Section 5 we conclude our paper.

## 2 Deterministic Fluid Model for Chunk Sharing

In this and next sections, we analyze the population dynamics of the sharing of a single chunk of a file. We study how the number of downloaders and seeds evolves over time from the emergence of the chunk to the disappearance of it. The disappearance of a single chunk means the death of the whole file sharing process since the file is not entire anymore. The work is motivated by the model of [6] but has some differences. In paper [6] the problem of sharing of several chunks concurrently is solved by assuming that peers can forward the chunks with a constant rate. However, we find the assumption unrealistic and the model probably hides some details of the population dynamics. For this reason we consider the sharing of a single chunk at a time. In addition, among others, papers [5] and [6] assume that at least one seed stays in the system keeping the chunks available. However, measurements of BitTorrent show that the file sharing process dies sooner or later [3]. Therefore the life time of the process is also studied.

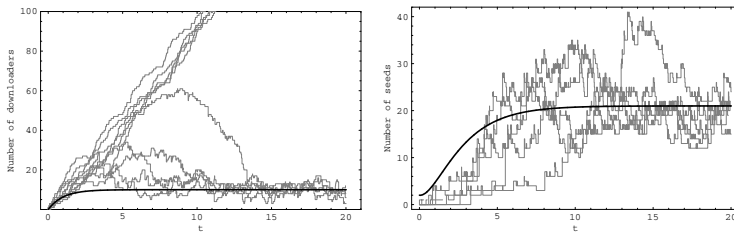
In the model, new requests for a chunk are assumed to arrive at the system with rate  $\lambda$  according to the Poisson process. The downloader can download the

file with rate  $\mu_d$ . On the other hand, the maximum upload rate of a peer for the chunk is assumed to be  $\mu_s$ . After the download, the status of the downloading peer changes from a *downloader* to a *seed* and the peer can distribute the chunk forward. Note that in this context, a peer is referred to as the seed if it has the chunk in question, but not necessarily all chunks of the file. The seed leaves the system with the probability  $\gamma$  per time unit. Let  $x(t)$  be the number of downloaders and  $y(t)$  be the number of seeds at time  $t$ . In the next sections we study the evolution of the pair  $(x, y)$  both by a deterministic fluid model but by a Markov model as well.

We consider a system where a peer starts to spread a single chunk to other peers that are willing to download it. If  $\mu_d x(t) < \mu_s y(t)$ , the downloaders can not use all service capacity provided by the peers. On the other hand, when  $\mu_d x(t) > \mu_s y(t)$  the upload capacity of seeds limits the download process. Thus the total service rate of the system is  $\min\{\mu_d x(t), \mu_s y(t)\}$ . First we construct a deterministic fluid model for the number of downloaders  $x(t)$  and seeds  $y(t)$ :

$$\begin{aligned} \frac{dx(t)}{dt} &= \lambda - \min\{\mu_d x(t), \mu_s y(t)\}, \\ \frac{dy(t)}{dt} &= \min\{\mu_d x(t), \mu_s y(t)\} - \gamma y(t), \end{aligned} \tag{1}$$

where  $y(0) = 1$  and  $x(0) = 0$ . Let  $\bar{x}$  and  $\bar{y}$  be possible equilibrium values of  $x(t)$  and  $y(t)$ . If  $\mu_d \bar{x} \leq \mu_s \bar{y}$ , the steady state solution is  $\bar{x} = \lambda/\mu_d$  and  $\bar{y} = \lambda/\gamma$ . From the constraint  $\mu_d \bar{x} \leq \mu_s \bar{y}$  we obtain the condition for the equilibrium:  $\mu_s \geq \gamma$ . When  $\mu_s < \gamma$  the solution of the equations (1) is  $\bar{y} = 0$  and  $\bar{x} \rightarrow \infty$ .



**Fig. 1.** The number of downloaders on the left side and the number of the seeds on the right side as a function of time (in units of  $1/\mu_d$ ). Solid lines: fluid model (1), gray lines: simulation.  $\lambda/\mu_d = 10$ ,  $\lambda/\gamma = 20$ ,  $\mu_s \geq \gamma$ .

The evolution of the number of downloaders and seeds is depicted in Figure 1. We have fixed  $\lambda/\mu_d$  and  $\lambda/\gamma$  to moderately small values in order to better demonstrate the dynamics of the system. The solid line corresponds to the solution of fluid model (1) and the gray lines to 10 different simulations. We can see that in the beginning the capacity of the system is not sufficient to serve chunk requests. This is seen as a dramatic increase in the number of downloaders. However, after some downloaders have changed their status to seeds, the system stabilizes. At the end time ( $t = 20$ ) 4 simulated processes of 10 have become extinct and the number of downloaders in those processes increases without any limit.

### 3 Markov Chain Model for Chunk Sharing

The deterministic fluid model of the previous subsection describes the average behavior of the sharing of the chunks. However, from the simulation results we saw two effects in the population dynamics that were not captured by the fluid model. First, when the chunk became available the seeds could not serve all the downloaders, and second, if the original seed can leave the system, the death of the chunk and the whole file sharing process is irrevocable, even if  $\mu_s > \gamma$ . The limited life span of the file sharing process has an influence on the performance of the system and has to be analyzed by some other models. To this end, in this subsection we study the evolution of the process  $(x, y)$  in more detail by a Markov chain model with absorption. We construct a continuous time Markov chain process, where the state is the pair  $(x, y)$  and the transition rate matrix is  $Q$  with the elements:

$$\begin{aligned} q((x, y), (x + 1, y)) &= \lambda, \\ q((x, y), (x - 1, y + 1)) &= \min\{\mu_d x, \mu_s y\}, \quad \text{if } x > 0, \\ q((x, y), (x, y - 1)) &= \gamma y, \quad \text{if } y > 0. \end{aligned} \quad (2)$$

The states  $(x, y)$  with  $y = 0$  in the Markov chain are absorbing states. Since we are not interested in the process after entering one of the absorbing states, we combine all of them into one state 0. The mean time to absorption can be determined as follows: Let  $b_i$  denote the mean time to absorption, when the system starts from state  $i$ . Given the transition matrix  $Q$ , the mean times to absorption  $b_i$  are determined by a familiar Markovian recursion:

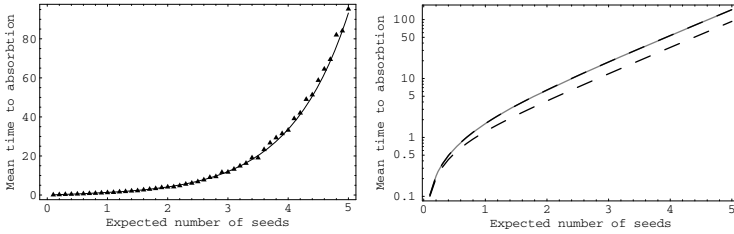
$$b_i = \frac{1}{q_i} \left( 1 + \sum_{j \neq i} q_{i,j} b_j \right), \quad (3)$$

where  $b_0 = 0$  and  $q_i = \sum_{i \neq j} q_{ij}$ . The absorption time starting from the initial state  $(0, 1)$ , i.e. the life time of the system, as a function of  $\lambda/\gamma$  is shown in the left side of Figure 2. The solid line is calculated by solving the set of linear equations (3) numerically in a truncated state space of  $35 \times 35$  states. The dots are obtained from simulation of the corresponding infinite system verifying the analytical results. The figure shows that the system life time increases exponentially as a function of the expected number of the seeds  $\lambda/\gamma$  in the system.

In one limit case the absorption time can easily be approximated. When the mean service times  $1/\mu_s$  and  $1/\mu_d$  are very small, the system can be modelled as an M/M/ $\infty$ -queue with arrival rate  $\lambda$  and departure rate  $\gamma$ . The mean time to absorption equals the average length of the busy period  $E[B]$  of M/M/ $\infty$ -queue:

$$E[B] = \frac{1}{\lambda} (e^{\lambda/\gamma} - 1). \quad (4)$$

The approximation and the analytical result from the Markov model are depicted on logarithmic scale on the right side of Figure 2. For  $\mu_s = \mu_d = 100$  the approximation coincides with the analytical result.



**Fig. 2.** Left figure: The mean time for absorption (in units of  $1/\lambda$ ), solid line: analytical results, dots: simulation. Right figure: Analytical results for  $\mu_s = \mu_d = 1$  (dashed line) and  $\mu_s = \mu_d = 100$  (upper line) and approximation (upper line, overlapping with  $\mu_s = \mu_d = 100$  line).

### 4 Location-Based Chunk Sharing Model

Our next objective is to study the possible reduction in network resource usage by a location-based peer selection policy, as opposed to random selection of the seed. We analyze location-aware sharing of a single chunk in a simplified setting where the underlying topology of the network is eliminated and replaced by a simple geometrical structure. By this approach we are able to estimate the capacity usage analytically.

As before, new requests for a chunk arrive in the system with rate  $\lambda$  according to the Poisson process. Each new request is associated with a peer  $i$ , whose location is assumed to be randomly chosen on the surface of a sphere following uniform distribution. We have chosen the spherical geometry primarily because it is symmetrical and has no artificial boundaries. It is also a natural choice if one considers a global network. Let  $R$  be the radius of the sphere and let the location of peer  $i$  be described by cylindrical coordinates  $z_i$  and  $\phi_i$ . It is easily verified that if  $z_i = -R + 2Ru$  and  $\phi_i = 2\pi u'$ , where  $u$  and  $u'$  are drawn from the uniform distribution  $U(0, 1)$ , the peers are uniformly located on the sphere.

Let  $D(t)$  be the set of downloaders and  $S(t)$  be the set of seeds at time  $t$ . Let parameter  $p_i$  denote the selected seed  $j$  of downloader  $i$ . As a metric for distance between two peers  $i$  and  $j$  we use the shortest path between the peers on the surface of the sphere, denoted by  $d_{i,j}$ .

How much downloading a chunk consumes resources of the underlying network is assumed to be proportional to the distance between the peers exchanging chunks. If the peers are far apart, transferring a chunk typically needs more links than in the case of two close peers. Let  $c(t)$  denote the total instantaneous capacity required for sharing chunks at time  $t$ ,  $c(t) = \sum_{i \in D(t), j = p_i} d_{i,j}$ , i.e.,  $c(t)$  describes the sum of distances between the peers sharing the chunk. However, when we consider the resource usage optimization, a more interesting quantity is the average capacity usage  $C$  per downloaded chunk over time period  $[t_0, t_{max}]$  defined as  $C = \frac{1}{n} \int_{t_0}^{t_{max}} c(t) dt$ , where  $n$  is the number of the chunks transferred within this period.

We consider two different peer selection policies: *Random peer selection* (RPS) policy, where the seed for download is selected randomly among all available peers, and *nearest peer selection* (NPS) policy, where the nearest possible peer in terms of the distance between the peers is selected.

#### 4.1 Analytical Bounds for Capacity Usage

In RPS, each downloader selects one random seed. The distance to a random seed is independent of the number of seeds. Assuming, without loss of generality, that the mean download time  $1/\mu_d$  of a chunk is one, the expected resource usage per a downloaded chunk is equal to the average distance between two points on a sphere (assumed to have unit area):  $E[C] = E[d] = \sqrt{\pi}/4$ .

In NPS, the nearest peer among  $y(t) + 1$  seeds is selected for download. If  $N$  points are randomly distributed on a sphere with unit area, the expected distance to the nearest neighbor can easily be determined,

$$E[d|N = n] = \frac{\Gamma(n - \frac{1}{2})}{2\Gamma(n)} = \frac{\sqrt{\pi}}{2} \prod_{i=0}^{n-2} \frac{i + \frac{1}{2}}{i + 1}, \quad (5)$$

which is very accurately approximated by  $E[d|N = n] \approx \frac{1}{2\sqrt{n-0.73}}$ , with a maximum error of only 0.16% occurring at  $n = 4$ . At time  $t$ ,  $N$  includes  $y(t)$  seeds and the downloader itself, meaning that  $N = y(t) + 1$ . The expected resource usage for NDP policy is:

$$E[C] = \sum_{y=0}^{\infty} p\{Y = y\} E[d|N = y + 1]. \quad (6)$$

In general, the steady state distribution of  $y(t)$ ,  $p\{Y = y\}$ , can be calculated from the Markov model of section 3. Due to complexity of the model, the solution cannot be expressed in a closed form. However, in a case where the service is always constrained by download rate and at least one peer stays in the system, the system of downloaders and seeds can be considered as two consecutive M/M/ $\infty$  queues, where arrival rates to the first and second queues are  $\lambda$  and the service rates are  $x\mu_d$  and  $y\gamma$ , respectively. It is well known that under these assumptions the steady-state distribution of the downloaders and the seeds follows the Poisson distribution. The expected resource usage is then:

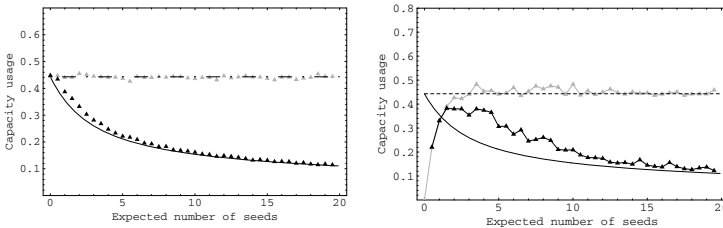
$$E[C] = \sum_{y=0}^{\infty} \frac{(\frac{\lambda}{\gamma})^y}{y!} e^{-\frac{\lambda}{\gamma}} \frac{\sqrt{\pi}}{2} \prod_{i=0}^{y-1} \frac{i + \frac{1}{2}}{i + 1}. \quad (7)$$

Note that this analytical value for capacity usage assumes that every time when the status of the seeds changes, the downloaders have to update their peers, and seek the closest peer again. This is, however, not very realistic. For this reason (7) can be viewed as a lower bound for the resource usage. Our simulations, however, suggest that this bound is not far from the resource usage of a more realistic peer selection scheme.

### 4.2 Simulation Results

Next we study by numerical examples how the selected policy affects the capacity usage. First, on the left side of Figure 3 we study the scenario explained in the previous subsection, where service is always constrained by download rate and at least one peer stays in the system. The capacity usage  $C$  is shown as a function of the expected number of seeds  $\lambda/\gamma$  (simulation starts at time 0,  $t_0 = 1000$  and  $t_{max} = 10000$ ). Gray triangles correspond to a simulation with RPS policy and black triangles to NPS policy. When  $\lambda/\gamma$  is small, seeds leave the system shortly after the download and the peers that want to download the chunk have to request it from the original seed. The distances from a downloader to the original seed using the two different policies are then the same. When  $\lambda/\gamma$  increases the number of seeds also increases and the selected policy has an effect on the resource usage. We can see that, e.g., for  $\lambda/\gamma = 20$  the capacity usage of the policy NPS is only 23 % of the capacity usage of the policy RPS. Simulation results are very close to analytical bounds, especially when  $\lambda/\gamma > 5$ .

Then we consider a system in which the service capacity is constrained by both upload and download rate and the system dies if all the seeds have left the system (model introduced in Section 2). When a new downloader arrives, it seeks either a random available (RPS) or the closest available (NPS) seed for download. The simulated average capacity usage per downloaded chunk over the period from 0 to the time of extinction for random and nearest peer selection policies are shown on the right side of Figure 3. For small  $\lambda/\gamma$  the system's life time is very short, and therefore we have done  $K = 1000/(\lambda/\gamma)$  simulations to ensure that also for small  $\lambda/\gamma$  we have enough simulations. When  $1/\gamma \ll 1/\lambda$ , after the arrival of the first downloader, most probably the system absorbs very shortly without any completed download. Considering only those simulation traces with at least one served peer distorts the realized service time of the accepted simulations close to zero. For this reason the capacity usage is also very small when  $1/\gamma \ll 1/\lambda$ . When  $\lambda/\gamma > 1$ , the realized service time is closer to the expected value  $1/\mu$  and the capacity usage is also greater. Finally, when the expected number of seeds



**Fig. 3.** Mean capacity usage as a function of  $\lambda/\gamma$ ,  $\mu_d = \mu_s = 1$ . Gray triangles: simulation of RPS, black triangles: simulation of NPS. Dashed line: Expected resource usage for RPS policy and solid line: Expected resource usage for NPS policy. Left figure: download-constrained system. Right figure: The service capacity constrained by both upload and download rate.



$\lambda/\gamma$  increases even more, the system most probably reaches its steady state before going to extinction. Also the capacity usage is then close to analytical bound calculated for the download-constrained system presented in the previous subsection.

## 5 Conclusion and Further Research Directions

In this paper we have studied the population dynamics of sharing of a single chunk in a P2P file sharing system. We have constructed a deterministic fluid model to analyze the evolution of the number of downloaders and seeds. The life time of the system is calculated by solving the absorption times of the corresponding Markov chain. We can see that the time to extinction increases exponentially as a function of the expected number of seeds in the system. Most important, we have proposed a spatio-temporal model to analyze the resource usage of the system. The analytical bounds for two different peer selection policies are derived. We find that by the peer selection policy where the closest peer is selected for download the resource usage of the network can be reduced to a fraction of the usage of random selection.

The arrival rate of new downloaders  $\lambda$  hardly remains constant over time. When a new file is released, demand for that is high adding the flash crowd effect but after some weeks or so it will be fade out. This affects the life cycle of the file sharing process. We plan to assess this issue in a future work.

In this paper we have considered the distribution of only a single chunk independently of other pieces. In the future, the model will be extended to capture the dynamics of multiple chunks as well.

## References

1. B. Cohen, Incentives Build Robustness in BitTorrent, 2003, <http://www.bittorrent.com/bittorrentecon.pdf>.
2. M. Izal, G. Uvroy-Keller, E.W. Biersack, P.A. Felber, A.Al Hamra, and L. Garcés-Erice, Dissecting BitTorrent: Five Months in a Torrent's Lifetime, PAM, 2004.
3. J.A. Pouwelse, P. Garbacki, D.H.J. Epema, H.J. Sips, The BitTorrent P2P File-sharing system: Measurements and analysis, IPTPS, 2005.
4. L. Massoulié and M. Vojnović, Coupon replication Systems, SIGMETRICS, 2005.
5. X. Yang, G. de Veciana, Service Capacity of Peer to Peer Networks, INFOCOM 2004.
6. D. Qiu, R. Srikant, Modeling and Performance Analysis of BitTorrent-Like Peer-to-Peer Networks, SIGCOMM 2004.
7. K.K. Ramachandran, B. Sikdar, An Analytic Framework for Modeling Peer to Peer Networks, INFOCOM 2005.