Extinction Probability in Peer-to-Peer File Diffusion

Sophie Hautphenne[1], Kenji Leibnitz[2], Marie-Ange Remiche[3]

1 Introduction

Recent measurement studies [8] have shown that peer-to-peer (P2P) file sharing applications are the major traffic source in the Internet. P2P applications, such as eDonkey, Kazaa, or BitTorrent, form overlay networks on the application layer and offer its peers to download and share their files with other peers in a highly distributed way. As a consequence, peers act simultaneously as both clients and servers. For a comprehensive survey of P2P technology, we refer to [7].

In this work, focus is put on characterizing the diffusion behavior of a single file shared by multiple peers in a hybrid P2P architecture (e.g. eDonkey [2]). This is done by means of a particular class of Markov Binary Tree (MBT) that we define in Section 2. We derive the probability that the sharing process of a file finishes given the probability that a peer stops sharing the file. A computational algorithm is proposed and its efficiency is compared to the order algorithm developed in [1] for MBTs.

2 File Diffusion Process

We consider a network of peers with a very large number N of connected peers and we assume that N is actually infinite. This assumption is justified since it was shown in [5] from measurements that about 50% of the total number of eDonkey users are connected to the seven largest index servers with up to 500,000 peers. In the eDonkey network, each file is logically split up into chunks with maximal size of 9.28MB, each made up of 180kB blocks which are downloaded individually.

A peer is said to be *active* in the system as long as it downloads or shares the file. Therefore, we have two periods of activity for a peer: the *downloading period* followed possibly by a *sharing period*.

When a peer wishes to download a file, it connects to a special node in the eDonkey network, the *index server*, and issues a query for that file. Since the search is performed by the index server as a centralized unit, eDonkey is referred as a hybrid P2P system, in contrast to Gnutella, for instance, where file searching is performed in an entirely distributed manner. After receiving a list of peers that are sharing the file from the index server, the downloading peer contacts each of these peers and places a request in their upload queues. After all blocks of a chunk have been received correctly, it is automatically shared to other peers. Finally, after all chunks of a file are complete, the user can continue sharing the file with other peers or selfishly remove it from share.

We assume in our model that only one transmission can be initiated at a time but several on-going transmissions can be achieved in parallel. Both downloading and sharing periods are random. At any time, a peer is free to remove the file from sharing, with probability $1 - p_{share}$. Furthermore, we assume that all peers are homogeneous and symmetric in access speed, thus, their download and upload rates are equal.

In our model, downloading and sharing periods are governed by two particular Markov phase processes. For the downloading period, the state-space is $\{0', 0, 1, \dots, n\}$, with n finite and where both 0 and $0'$ are absorbing states. A peer is in state $0'$ when it stops sharing the file, and in state 0 when a transmission to another peer is initiated, otherwise the peer is in some state $i \in \{1, \dots, n\}$. The initial probability vector of the phase process is $(0, 0, \underline{\tau}_1)$, where $\underline{\tau}_1$ is a row vector of size n and its infinitesimal generator is

$$Q_A = \begin{pmatrix} 0 & 0 & \mathbf{0} \\ 0 & 0 & \mathbf{0} \\ (1 - p_{share})\boldsymbol{a} & p_{share}\boldsymbol{a} & A \end{pmatrix}$$

where A is an $n \times n$ matrix and $\boldsymbol{a}$ is a column vector of size n, with $\boldsymbol{a} + A\boldsymbol{e} = \mathbf{0}$, where $\boldsymbol{e}$ is a column vector of ones of appropriate size. The distribution of the time till absorption into any of the absorbing states is Phase-type (see Latouche and Ramaswami [6] Chapter 2).

We start the diffusion process at time 0 with one initial peer that is downloading the file. After completion of the transfer, the peer is either in state $0'$ or in state 0. In the latter case, the sharing period starts. The sharing period is modeled as a terminating Phase-type renewal process. The intervals between renewals are governed by a Markov process defined on states $\{0'', n+1, \dots, n+m\}$, with m finite, initial probability vector $(0, \underline{\tau}_2)$ and infinitesimal generator

$$Q_B = \begin{pmatrix} 0 & \mathbf{0} \\ \boldsymbol{b} & B \end{pmatrix},$$

with $\boldsymbol{b} + B\boldsymbol{e} = \mathbf{0}$. At a renewal epoch, if the peer is still sharing the file (with probability p_{share}), a new peer begins to download the file and a new renewal interval begins;

[1]Université Libre de Bruxelles, Département d'Informatique, CP 212, Blvd du Triomphe, B-1050-Bruxelles, Belgium, e-mail: shautphe@ulb.ac.be
[2]Osaka University, Graduate School of Information Science and Technology, 1-5 Yamadaoka, Suita, Osaka 565-0871, Japan, e-mail: leibnitz@ist.osaka-u.ac.jp
[3]Université Libre de Bruxelles, Faculté des Sciences Appliquées, SMG - CP210/01, Bld du Triomphe, B-1050 Bruxelles, Belgium, e-mail: mremiche@ulb.ac.be

otherwise, the peer has removed itself from share and is no longer active. We call the renewal epochs *branching points*.

The life of a single peer is thus governed by a transient Phase-type renewal process with initial phase distribution $\boldsymbol{\tau}_1 = (\underline{\tau}_1, \mathbf{0}')$ and infinitesimal generator

$$T = \begin{pmatrix} A & p_{share}\boldsymbol{a} \otimes \underline{\tau}_2 \\ 0 & B \end{pmatrix},$$

the phase distribution after a renewal is $\boldsymbol{\tau}_2 = (\mathbf{0}', \underline{\tau}_2)$, the rate of termination (of stopping to share the file) is $\boldsymbol{d} = ((1 - p_{share})\boldsymbol{a}', (1 - p_{share})\boldsymbol{b}')'$, while that of a new renewal is $\boldsymbol{t}$ with $\boldsymbol{t} = (\mathbf{0}', p_{share}\boldsymbol{b}')'$.

The whole file diffusion process is a Markov Binary Tree in which the growth of the branches is governed by a phase-type process with generator T. At a branching point, the branch may die out, if the process is no longer sharing, or it may divide into two arcs, of which the initial phases are independently chosen with the distribution $\boldsymbol{\tau}_1 = (\underline{\tau}_1, \mathbf{0}')$ for the left (or child) branch, this corresponds to a new peer that asks for downloading, and $\boldsymbol{\tau}_2 = (\mathbf{0}', \underline{\tau}_2)$ for the right (or parental) branch, which corresponds to the peer which is already in the sharing mode.

This splitting mechanism is captured in the $n \times n^2$ birth rates matrix R given by $R = \boldsymbol{t}(\boldsymbol{\tau}_1 \otimes \boldsymbol{\tau}_2)$. The entry $R_{i;j,k} = t_i\,(\boldsymbol{\tau}_1)_j\,(\boldsymbol{\tau}_2)_k$ gives the rate with which a branch in phase i splits into two parts to give birth to a child arc in phase j and a parental arc in phase k.

The measure of interest is the extinction probability, that is, the probability that the diffusion of the files will ever completely stop. In the next section, we propose to derive an efficient algorithm with that objective.

3 The Thicknesses Algorithm

The MBT constructed in Section 2 can be seen as an exponential continuous-time branching process with $n + m$ types: one only has to see the phases of the renewal process as the types of the branching process. At phase transition times, the peer may (i) stop the sharing, (ii) continue its activity, but in another phase, (iii) share the file to another peer. This behavior is recorded in the progeny generating function. The progeny generating function of a type-i peer is given by

$$P_i(s) = \frac{d_i}{-T_{ii}} + \sum_{k=1,k\neq i}^{n+m} \frac{T_{ik}}{-T_{ii}}\,s_k + \sum_{k,j=1}^{n+m} \frac{R_{i;j,k}}{-T_{ii}}\,s_k\,s_j, \quad (1)$$

where $\boldsymbol{s} = (s_1, s_2, \ldots, s_{n+m})$ are complex numbers whose modulo is less or equal than 1.

The vector $\boldsymbol{q}$ of extinction probabilities, given the initial phase of the MBT, satisfies the system $\boldsymbol{P}_i(\boldsymbol{q}) = \boldsymbol{q}$ (as proved in Harris [3]), which, by (1), may be written as

$$\boldsymbol{q} = \boldsymbol{\theta} + \Psi(\boldsymbol{q} \otimes I)\,\boldsymbol{q}, \quad (2)$$

where $\boldsymbol{\theta} = (-T)^{-1}\,\boldsymbol{d}$ and $\Psi = (-T)^{-1}\,\boldsymbol{t}(\boldsymbol{\tau}_1 \otimes \boldsymbol{\tau}_2)$.

The thicknesses algorithm developed in Hautphenne [4] goes as follows: define $\boldsymbol{q}_2(0) = \boldsymbol{\theta}$ and

$$\begin{aligned} \boldsymbol{q}_1(2k-1) &= \{I - \Psi\,[I \otimes \boldsymbol{q}_2(2k-2)]\}^{-1}\boldsymbol{\theta} \\ \boldsymbol{q}_2(2k) &= \{I - \Psi\,[\boldsymbol{q}_1(2k-1) \otimes I]\}^{-1}\boldsymbol{\theta} \end{aligned}$$

for $k \geq 1$. Each stage of the algorithm computes an approximation $\boldsymbol{q}_1$ or $\boldsymbol{q}_2$ of the vector $\boldsymbol{q}$. This algorithm is based on the notion of left and right thickness of a tree, which is introduced in [4] and does converge to the vector $\boldsymbol{q}$.

4 Conclusion

We show through its interpretation and through numerical illustration that our algorithm, is more efficient than the order algorithm, in cases where trees have long branches on both the left and the right sides or in cases where the total number of branching points is big compared to the activity duration of a single peer.

This first P2P diffusion model needs to be extended in order to take into account the number of parallel ongoing transmissions at a same peer, but also to include the downloading and sharing of chunks as defined in [5].

References

[1] N.G. Bean, N. Kontoleon, and P.G. Taylor. Algorithms for determining the probability of eventual extinctio= n of a Markovian binary tree. Technical report, University of Adelaide & University of Melbourne, 2004.

[2] eDonkey2000. http://www.eDonkey2000.com/.

[3] T.E. Harris. *The Theory of Branching Processes*. Dover, New-York, 1963.

[4] S. Hautphenne. *Processus de branchement et modèles de phases*. Mémoire de fin d'études, Université Libre de Bruxelles, Belgium, June 2005.

[5] T. Hoßfeld, K. Leibnitz, R. Pries, K. Tutschku, P. Tran-Gia, and K. Pawlikowski. Information diffusion in edonkey filesharing networks. In *Australian Telecommunication Networks and Applications Conference (ATNAC 04)*, Sydney, Australia, 2004.

[6] G. Latouche and V. Ramaswami. *Introduction to Matrix Geometric Methods in Stochastic Modeling*. ASA-SIAM Series on Statistics and Applied Probability. SIAM, Philadelphia, PA, 1999.

[7] E. Keong Lua, J. Crowcroft, M. Pias, R. Sharma, and S. Lim. A survey and comparison of peer-to-peer overlay network schemes. *IEEE Surveys and Tutorials*, March 2004.

[8] K. Tutschku. A measurement-based traffic profile of the eDonkey filesharing service. In *5th Passive and Active Measurement Workshop (PAM2004)*, Antibes Juan-les-Pins, France, Apr. 2004.