Modeling of P2P File Sharing with a Level-Dependent QBD Process

Sophie Hautphenne[a], Kenji Leibnitz[b], and Marie-Ange Remiche[c]
[a] *Université Libre de Bruxelles, Département d'Informatique, CP 212, Blvd du Triomphe, B-1050-Bruxelles, Belgium,* shautphe@ulb.ac.be
[b] *Osaka University, Graduate School of Information Science and Technology, 1-5 Yamadaoka, Suita, Osaka, 565-0871 Japan,* leibnitz@ist.osaka-u.ac.jp
[c] *Université Libre de Bruxelles, Faculté des Sciences Appliquées, Code SMG - CP 165/15, Av. F.D. Roosevelt, 50, B-1050-Bruxelles, Belgium,* mremiche@ulb.ac.be

Abstract

In this paper we propose to analyze a peer-to-peer (P2P) file sharing system by means of a so-called level-dependent Quasi-Birth-and-Death (QBD) process. We consider the dissemination of a single file consisting of different segments and include a model for the upload queue management mechanism with peers competing for bandwidth. By applying an efficient matrix-analytic algorithm we evaluate the performance of P2P file diffusion in terms of the corresponding extinction probability, i.e., the probability distribution that the sharing process ends.

Keywords: peer-to-peer, branching process, matrix analytic methods, logarithmic-reduction algorithm.

I. Introduction

With the introduction of *peer-to-peer* (P2P) technology in networks for file-sharing and content distribution, the volume of transported traffic has recently enormously increased. The nodes participating in the P2P network are called *peers* and form logical overlay structures on application layer above the IP topology. One of the main advantages of using P2P networks for content distribution is their higher scalability to a growing number of file requests, especially in the presence of flash crowd arrivals [12]. Unlike conventional client/server architectures, all peers act simultaneously as clients and servers, thus, shifting the load from a single server to several peers sharing a specific file. Additionally, since the source of a file is no longer stored at a single location, the P2P network is more robust to failures.

However, there are also certain dangers on entirely relying on P2P networks for file distribution. Firstly, the data is no longer kept at a single trusted source, as each peer which hosts the file may modify the data willingly or unwillingly, thus, causing the distribution of corrupt information. This is referred to as *poisoning* or *pollution* [3]. Secondly, the existence of a sharing peer in the network can not be guaranteed due to *churn*, i.e., the process of peers entering and leaving the network. The sharing of files is controlled by the peers' behavior (willingness to share after downloading, patience, etc.) and they may arbitrarily join or leave the network at any instant. If the peer, which has the last part of the file, leaves the network, this information is lost and other peers can no longer retrieve this data. For this reason, specific P2P architectures like Chord [14] employ mechanisms to maintain a certain number of replicas in the network.

In this work we study the probability that the diffusion of a file will eventually come to a halt in an unstructured P2P file sharing network, which we define as the *extinction* of the file. We extend our previous model in [5] where we used a *Markovian Binary Tree* (MBT) to model the file sharing network and we formulated an algorithm to compute the extinction probability. However, the previous model only considered the sharing of entire files. In this paper, we extend the model to include parts of the file being shared to show a more accurate sharing behavior. This is achieved by using a level-dependent *Quasi-Birth-and-Death* (QBD) process. By adapting the logarithmic-reduction algorithm (see Latouche and Ramaswami [7]), we actually compute the probability that file diffusion ends due to the lack of peers sharing a part of the file.

This paper is organized as follows. First, we will briefly summarize related work on modeling of P2P file sharing mechanisms for content distribution in Section II. This is followed by the formulation of our basic assumptions on the file sharing network in Section III. We will not specifically focus on any existing P2P protocol, but use a rather generic specification which roughly resembles the eDonkey protocol. In Section IV we will formulate two analytical models corresponding to two different systems in which either the sharing process stops when the entire file is lost, or when any of the segments is missing. Accordingly, we construct the corresponding level-dependent QBD process and we develop algorithms necessary to obtain the extinction probability in both settings. Finally, we provide some numerical results on the impact of the system parameters on the performance of the system in Section V.

II. Related Work

Most studies on the evaluation of the performance of P2P systems as content distribution networks rely on measurements or simulations of existing P2P networks. For example, Saroiu *et al*. [13] conducted measurement studies of content delivery systems that were accessed by the University of Washington. The authors distinguish between traffic from P2P, WWW, and the Akamai content distribution network, and they found that the majority of volume is transported over P2P. Hoßfeld *et al*. [6] provide a simulation study of the well-known eDonkey network and examine the file diffusion properties under constant and flash crowd arrivals.

An analytical model for performance evaluation of a generalized P2P system is given by Ge *et al*. [4]. On the other hand, other published work consider specific existing network types. For example, Qiu *et al*. [10] use a fluid model for BitTorrent and investigate the performance in steady state. They studied the effectiveness of the incentive mechanism in BitTorrent and prove the existence of a Nash equilibrium. Rubenstein and Sahu [12] mathematically show that unstructured P2P networks have good scalability and are well suited to cope with flash crowd arrivals. A fluid-diffusive P2P model from statistical physics is presented by Carofiglio *et al*. [2]. Both, the user and the content dynamics are included, but this is only done on file level and without pollution. These studies show that by providing incentives to the peers for sharing a file, the diffusion properties are improved. Yang and de Veciana [15] investigate the service capacity of P2P networks by considering two models, one for the transient state with flash crowds and one in steady state.

Christin *et al*. [3] measured content availability of popular P2P file sharing networks and used this measurement data for simulating different pollution and poisoning strategies. They show that only a small number of fake peers can seriously impact the user's perception of content availability. In [8] a diffusion model for modeling eDonkey-like P2P networks is presented based on a model from mathematical biology. This model includes pollution and a peer patience threshold at which it aborts its download attempt and retries later again. It is shown that an evaluation of the diffusion process is not accurate enough when steady state is assumed or the model only considers the transmission of the complete file, especially in the presence of flash crowd arrivals. That model is extended in [9] to analytically compare the performance of P2P file sharing networks to that of client/server systems.

III. Peer-to-Peer File Sharing Model

Let us now define the assumptions we make on the P2P file sharing model in this paper. We assume an unstructured P2P network operating similar to the eDonkey network. However, the model is not restricted to eDonkey, but can in fact be applied to other file sharing networks as well. The sharing of a

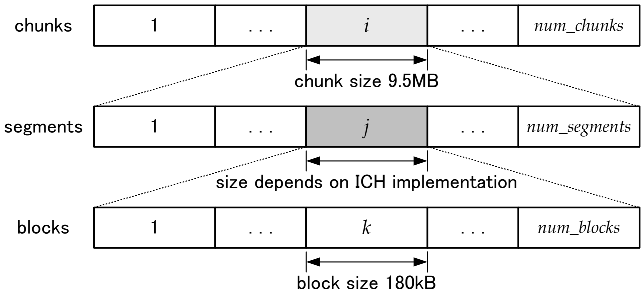

Fig. 1. File structure consisting of chunks, segments, and blocks

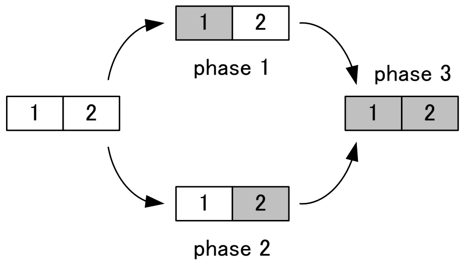

Fig. 2. Possible phases for downloading a chunk

file with size F is performed in units of *chunks*, which is further split into smaller units called *blocks*, see Figure 1. In eDonkey, a chunk has the size of 9.28 MB and a block is 180 kB. After each chunk has been downloaded, it is checked for errors and if the hash value is incorrect, all blocks of the chunk are discarded and downloaded again. After all chunks of a file have been successfully downloaded, the peer may decide if it keeps the file as a *seeder* in the network for other peers to download or if it is removed from sharing (*leecher* or *free rider*). In this work, we assume that the file consists of a single chunk, corresponding, for example, to a single mp3 audio file, as this is enough to capture the basic characteristics of the diffusion behavior.

A. Upload Queue Management

In order to manage the bandwidth for other peers requesting the file, an upload queue mechanism is maintained. A peer requests individual blocks from other peers sharing the chunk containing the desired block. All requests are appended to the waiting list of the sharing peer and a weighting mechanism handles the scheduling of the upload queue requests for transmission. The detailed procedure of the queue management takes several features into account that depend on the individual settings of the sharing peer like upload bandwidth and number of simultaneous uploads.

In the original version of eDonkey, error detection is done after all blocks of a chunk have been received and the complete chunk is discarded in the case of an error. However, this is not very effective and in more recent versions of eDonkey clients, e.g. eMule, the *Intelligent Corruption Handling* (ICH) mechanism is implemented that performs the error detection on smaller data units than chunks and which we define in the following as *segments*. Instead of discarding the complete chunk when at least one corrupted block is received, only all blocks of the damaged segment need to be re-requested. The specific size of a segment depends on the settings of the ICH mechanism.

With the model for the upload queue mechanism and corruption handling, it is sufficient to assume that a chunk only needs to be modeled consisting of few segments instead of individual blocks. In this study we assume that a chunk consists of two segments, i.e. $N_s = 2$ and the size of a segment is $Z = 4.64\,\mathrm{MB}$, the size F of the whole chunk being below or equal to $9.28\,\mathrm{MB}$.

B. Download Bandwidth

Let us define the upload and download rates as r_u and r_d, respectively. For the sake of simplicity, we use the same assumption as in [8] with homogeneous users with ADSL connections, resulting in rates of $r_u = 128\,\mathrm{kbps}$ and $r_d = 768\,\mathrm{kbps}$. Further let us denote the number of peers sharing a certain segment as S and the peers downloading it as D. Since eDonkey employs a fair share mechanism for the upload rates, there are on average S/D sharing peers for a single downloading peer and we multiply this value with r_u which gives us the bandwidth on the uplink. However, since the download bandwidth could be the limiting factor, the resulting effective transition rate consists of the minimum of both terms divided by the size of a segment Z. In the case of $N_s = 2$ segments, this results in the rates μ_1 and μ_2 given in Equation (1).

$$\mu_1(S, D) = \frac{1}{Z} \min \left\{ \frac{S}{D} r_u, r_d \right\} \qquad \mu_2(S, D) = \frac{1}{F - Z} \min \left\{ \frac{S}{D} r_u, r_d \right\} \tag{1}$$

IV. Analytical P2P Model

Let us consider a chunk to be made up of two segments: segment 1 and segment 2. This provides a three-phases system. A peer will be in phase 1 or 2 if it has only segment 1 or 2, respectively. If the peer has both segments (i.e. the complete chunk), it will be in phase 3, see Figure 2. New peers appear at random time in the system determined by an exponential random variable whose rate is defined by Equation (1) and by the current state of the system which is expressed by the number of peers S_i in each phase i. For the sake of simplicity we can assume that the rates at which a peer stops sharing a segment is independent of the segment number, and is equal to d.

Let us now define the stochastic process $\{(X(t), \boldsymbol{\varphi}(t))\}$, where $X(t)$ represents the total number of peers with segments 1 or/and 2 present in the system at time t, and $\boldsymbol{\varphi}(t) = (\varphi_1(t), \varphi_2(t), \varphi_3(t))$ denotes the number of peers in each phase present in the system at time t, with $\boldsymbol{\varphi}(t)\mathbf{1} = X(t)$. Here, $\mathbf{1}$ denotes a vector with ones.

We consider two views to measure the extinction probability of the file sharing process, an *optimistic* and a *pessimistic* view. In the optimistic view, we assume that the sharing process ends when no more segments are available in the system. In the pessimistic case, the file sharing process ends as soon as one of the two segments is missing. We call this event a *catastrophe*. We now differentiate the two models corresponding to the two views described above.

A. Level-Dependent QBD

In this first setting, recall that the sharing process ends when there is no more segment available in the system. The stochastic process $\{(X(t), \boldsymbol{\varphi}(t))\}$ is an *absorbing level-dependent quasi-birth and-death process*, of which the generator Q can be written as in Equation (2).

$$Q = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & \dots \\ A_2^{(1)} & A_1^{(1)} & A_0^{(1)} & 0 & 0 & 0 & \dots \\ 0 & A_2^{(2)} & A_1^{(2)} & A_0^{(2)} & 0 & 0 & \dots \\ 0 & 0 & A_2^{(3)} & A_1^{(3)} & A_0^{(3)} & 0 & \dots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix} \tag{2}$$

This process has been extensively studied in the past (see Latouche and Ramaswami [7] and references therein). In this setting, time to extinction of the system is clearly equal to the time until absorption. In the remainder of this section, we first elaborate on the content of the $A_i^{(j)}$ matrices and then give the algorithmic procedure in order to compute the absorption time in this level-dependent QBD with generator Q.

1) Level-Dependent QBD Generator Description: Let us recall that the state (S_1, S_2, S_3) means that we have S_1 peers in phase 1 (with only segment 1), S_2 peers in phase 2 (with only segment 2), and S_3 peers in phase 3 (with the complete file). We define the state sub-space $L(k)$ as

$$L(k) = \{(S_1, S_2, S_3) : S_1 \geq 0, S_2 \geq 0, S_3 \geq 0;\ S_1 + S_2 + S_3 = k\},$$

which gives all states of the system at level k, that is when k peers are present in the system and its cardinality is

$$|L(k)| = \frac{1}{2}(k+2)(k+1).$$

In the following, we take the lexicographic order to enumerate the states of each level.

When the system contains a single peer (that is, when its state is in $L(1)$), the peer may stop sharing the one segment it possesses with rate d or another peer may start downloading the segment. This latter occurs at a rate given by the matrix $A_0^{(1)}$. The transition matrices from the first level $A_2^{(1)}$ and $A_0^{(1)}$ are given by

$$A_2^{(1)} = \begin{bmatrix} d \\ d \\ d \end{bmatrix}$$
$$A_0^{(1)} = \begin{bmatrix} \mu_1(1, S_2+1) & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \mu_2(1, S_1+1) & 0 & 0 \\ 0 & 0 & \mu_1(1, S_1+S_2+1) & 0 & \mu_2(1, S_1+S_2+1) & 0 \end{bmatrix}.$$

For example, if the system is in state $(0,1,0)$, only a new peer with segment 2 may appear, i.e., the system is in state $(0,2,0)$. This happens at a rate $\mu_2(1, S_1+1)$, see Equation (1).

Usually, a peer may also change its phase (from 1 to 3 or from 2 to 3). Such a transition keeps the level at 1 since no new peer arrives in the system. However, if a peer in phase 1 (or phase 2) is alone in the system, it will not be able to download the missing segment and to change into phase 3. Thus, the transition rate from phase 1 (or from phase 2) to phase 3 is $\mu_i(0,1) = 0$ for $i = 1, 2$. The diagonal elements of $A_1^{(1)}$ (and of all $A_1^{(k)}$, $k \geq 2$) are such that $Q\mathbf{1} = \mathbf{0}$.

The possible transitions from a state $(S_1, S_2, S_3) \in L(k)$ with $k \geq 2$ are described below:

$A_2^{(k)}$: The system may lose a peer in phase i with rate d multiplied by the number of peers in phase i, that is S_i with $i = 1, 2, 3$.

$A_0^{(k)}$: The two possible transitions listed in the table below may be interpreted with a similar argument. For the first transition, two events may happen: either the new peer was downloading the segment 1 from one of the S_1 peers, or it received it from one of the S_3 peers.

Transitions	**Rates**
$(S_1, S_2, S_3) \to (S_1+1, S_2, S_3)$	$S_1\,\mu_1(1, S_2+1) + S_3\,\mu_1(1, S_1+S_2+1)$
$(S_1, S_2, S_3) \to (S_1, S_2+1, S_3)$	$S_2\,\mu_2(1, S_1+1) + S_3\,\mu_2(1, S_1+S_2+1)$

$A_1^{(k)}$: A peer in phase 1 changes into a peer in phase 3 with the rate $\mu_2(S_2+S_3, S_1)$, since S_1 peers are competing for the $(S_2+S_3) \times r_u$ available bandwidth. The same argument holds for a peer in phase 2 changing into a peer in phase 3. Let us recall that the diagonal elements are such that $Q\mathbf{1} = \mathbf{0}$.

Transitions	**Rates**
$(S_1, S_2, S_3) \to (S_1-1, S_2, S_3+1)$	$\mu_2(S_2+S_3, S_1)$
$(S_1, S_2, S_3) \to (S_1, S_2-1, S_3+1)$	$\mu_1(S_1+S_3, S_2)$
Diagonal element	Parameter of the exponential
$(S_1, S_2, S_3) \to (S_1, S_2, S_3)$	$-k\,d - S_1\,\mu_1(1, S_2+1) - S_2\,\mu_2(1, S_1+1)$ $-S_3\,\mu_1(1, S_1+S_2+1) - S_3\,\mu_2(1, S_1+S_2+1)$ $-\mu_2(S_2+S_3, S_1) - \mu_1(S_1+S_3, S_2)$

2) Probability of Extinction: Our interest lies in computing the probability that the sharing process in the particular system setting described in the previous section will terminate at some point. Let $\gamma(0)$ be the first time the system is in level 0, i.e., no more segments are available. Let $\boldsymbol{e}_i$ be a unit vector with a

1 at the i-th entry and 0 elsewhere. We define $(\boldsymbol{G}_1)_i$ as the probability that the system, starting in level 1 with $\boldsymbol{\varphi}(0) = \boldsymbol{e}_i$, will eventually reach level 0, that is

$$(\boldsymbol{G}_1)_i = P\left[\gamma(0) < \infty \,|\, \boldsymbol{\varphi}(0) = \boldsymbol{e}_i\right] \qquad i = 1, 2, 3.$$

It was proven in [11] that this vector is explicitly given by

$$\boldsymbol{G}_1 = \sum_{l=0}^{\infty} \left[\prod_{i=0}^{l-1} U_{2^i}^i\right] D_{2^l}^l \tag{3}$$

where the matrices U_k^l and D_k^l are given by the following recursive equations.

$$U_k^0 = \left(-A_1^{(k)}\right)^{-1} A_0^{(k)} \tag{4}$$

$$D_k^0 = \left(-A_1^{(k)}\right)^{-1} A_2^{(k)} \tag{5}$$

$$U_k^l = \left[I - U_k^{l-1} D_{k+2^{l-1}}^{l-1} - D_k^{l-1} U_{k-2^{l-1}}^{l-1}\right]^{-1} U_k^{l-1} U_{k+2^{l-1}}^{l-1} \qquad l \geq 1 \tag{6}$$

$$D_k^l = \left[I - U_k^{l-1} D_{k+2^{l-1}}^{l-1} - D_k^{l-1} U_{k-2^{l-1}}^{l-1}\right]^{-1} D_k^{l-1} D_{k-2^{l-1}}^{l-1} \qquad l \geq 1 \tag{7}$$

We use the logarithmic-reduction algorithm, adapted for level-dependent QBD in [11]. A nice interpretation of this algorithm, as presented in [11], exists that we now recall. The matrices $U_{2^i}^i$ and $D_{2^l}^l$ in (3) may be interpreted as

$$U_{2^i}^i = P\left[\gamma(2^{i+1}) < \gamma(0) \,\wedge\, \varphi(\gamma(2^{i+1})) \,|\, X(0) = 2^i\right]$$
$$D_{2^l}^l = P\left[\gamma(0) < \gamma(2^{l+1}) \,\wedge\, \varphi(\gamma(0)) \,|\, X(0) = 2^l\right]$$

where $\gamma(k)$ is the first passage time to level k, that is $\gamma(k) = \inf\{t \geq 0 : X(t) = k\}$ with $k \geq 0$. So, the l-th term of the sum in (3) has the following interpretation.

$$\left[\prod_{i=0}^{l-1} U_{2^i}^i\right] D_{2^l}^l = P[\gamma(2^l) < \gamma(0) < \gamma(2^{l+1}) \,\wedge\, \varphi(\gamma(0)) \,|\, X(0) = 1] \tag{8}$$

We clearly see that summing Equation (8) for $l = 0$ to infinity gives us the vector $\boldsymbol{G}_1$.

B. Level-Dependent QBD with Catastrophes

The model in the previous section considered that the file dissemination terminates when no more segments are available for sharing in the system. However, in reality when only an individual segment or an incomplete file remains in the network, no peer is able to completely retrieve the file anymore. Therefore, we now consider that a file is not available for sharing as soon as one of the segments is lost. In this case, the process ends in an absorbing state belonging to level $L(0)$ which is defined in this new setting as:

$$L(0) = \{(0,0,0), (n,0,0), (0,n,0);\, n \in \mathbb{N}_0\}$$

where $\mathbb{N}$ is the set of natural numbers. We propose to gather all of these states $\{(n,0,0);\, n \in \mathbb{N}_0\}$ and $\{(0,n,0);\, n \in \mathbb{N}_0\}$, respectively, into one state each labeled $(k,0,0)$ and $(0,k,0)$, respectively. The sub-space $L(0)$ is thus composed of three states, that is $\{(0,0,0), (k,0,0), (0,k,0)\}$. Other level state-spaces are

$$L(k) = \{(i,j,l) \,|\, i,j \in \mathbb{N}, l \in \mathbb{N}_0\,\} \cup \{\,(i,j,0) \,|\, i,j \in \mathbb{N}_0\} \qquad k \geq 1$$

where $i + j + l = k$. The time to extinction is thus equal to the time to absorption and the generator of this new level-dependent QBD is given in Equation (9).

$$Q = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & \cdots \\ A_2^{(1)} & A_1^{(1)} & A_0^{(1)} & 0 & 0 & 0 & \cdots \\ A_3^{(2)} & A_2^{(2)} & A_1^{(2)} & A_0^{(2)} & 0 & 0 & \cdots \\ A_3^{(3)} & 0 & A_2^{(3)} & A_1^{(3)} & A_0^{(3)} & 0 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix} \tag{9}$$

The rates of catastrophe, determined by $A_3^{(k)}$, are given by the transitions and corresponding rates.

Transitions	**Rates**
$S_2 > 0 : (0, S_2, 1) \to (0, k, 0)$	d
$S_2 > 0 : (1, S_2, 0) \to (0, k, 0)$	d
$S_1 > 0 : (S_1, 0, 1) \to (k, 0, 0)$	d
$S_1 > 0 : (S_1, 1, 0) \to (k, 0, 0)$	d
$(0, 0, 1) \to (0, 0, 0)$	d

Accordingly, matrix $A_2^{(k)}$ becomes as shown below.

Transitions	**Rates**
$S_1 > 1$ or $S_3 > 0 : (S_1, S_2, S_3) \to (S_1 - 1, S_2, S_3)$	$S_1\, d$
$S_2 > 1$ or $S_3 > 0 : (S_1, S_2, S_3) \to (S_1, S_2 - 1, S_3)$	$S_2\, d$
$(S_1 > 0$ and $S_2 > 0)$ or $S_3 > 1 : (S_1, S_2, S_3) \to (S_1, S_2, S_3 - 1)$	$S_3\, d$

The other transitions described in matrices $A_0^{(k)}$ and $A_1^{(k)}$ stay the same as previously described in Section IV-A1 for the first model.

The probability of absorption can now be computed by extending the results in [1]. Let $\boldsymbol{G}_0^{(k)}$ be a matrix whose (i, j)-th element is the probability that the process reaches level 0 for the first time in phase j, given that the process starts in phase i of level $k \geq 1$ and levels 1 to $k - 1$ are taboo. Let $\boldsymbol{G}_k$ be the matrix whose (i, j)-th element is the probability that the process reaches level $k - 1$ for the first time in phase j given that the process starts in phase i of level $k \geq 1$. The absorption probability is then given by $\boldsymbol{G}_1$ which is here also equal to $\boldsymbol{G}_0^{(1)}$ by definition of this quantity. Moreover, we have for $k \geq 2$ that G_k is given by

$$\boldsymbol{G}_k = \left(A_1^{(k)}\right)^{-1} A_2^{(k)} + \left(A_1^{(k)}\right)^{-1} A_0^{(k)}\, \boldsymbol{G}_{k+1}\, \boldsymbol{G}_k. \tag{10}$$

Starting from level k, the QBD may directly move to level $k - 1$ with probability $\big((A_1^{(k)}\big)^{-1} A_2^{(k)}$, or it may move up to level $k + 1$ with probability $\big(A_1^{(k)}\big)^{-1} A_0^{(k)}$. Upon arrival in level $k + 1$, it eventually returns to level k with probability $\boldsymbol{G}_{k+1}$ and then to level $k - 1$, with probability $\boldsymbol{G}_k$. However, the equation for $\boldsymbol{G}_1$ is slightly different and is given by

$$\boldsymbol{G}_1 = \left(A_1^{(1)}\right)^{-1} A_2^{(1)} + \left(A_1^{(1)}\right)^{-1} A_0^{(1)} \left[\boldsymbol{G}_2\, \boldsymbol{G}_1 + \boldsymbol{G}_0^{(2)}\right].$$

Indeed, if the process moves up to level 2 (the second term in this sum), then to reach level 0, it may first return to level 1, with probability $\boldsymbol{G}_2$, and then move to level 0 with probability $\boldsymbol{G}_1$. It may also be directly absorbed in level 0 this time without returning to level 1 first. This happens with probability

$\boldsymbol{G}_0^{(2)}$. Thus, to compute $\boldsymbol{G}_1$, we need to know $\boldsymbol{G}_2$ and $\boldsymbol{G}_0^{(2)}$. More generally, $\boldsymbol{G}_0^{(k)}$ satisfies the following recursive equation.

$$\boldsymbol{G}_0^{(k)} = \left(A_1^{(k)}\right)^{-1} A_3^{(k)} + \left(A_1^{(k)}\right)^{-1} A_0^{(k)} \left[\boldsymbol{G}_{k+1}\, \boldsymbol{G}_0^{(k)} + \boldsymbol{G}_0^{(k+1)}\right] \tag{11}$$

Its interpretation follows directly from the definition of $\boldsymbol{G}_0^{(k)}$ using the same argument as used before. Thus, writing $Q_i^{(k)} = \left(-A_1^{(k)}\right)^{-1} A_i^{(k)}$, we have explicitly

$$\boldsymbol{G}_0^{(k)} = \left[I - Q_0^{(k)}\, \boldsymbol{G}_{k+1}\right]^{-1} \left[Q_3^{(k)} + Q_0^{(k)}\, \boldsymbol{G}_0^{(k+1)}\right]. \tag{12}$$

This implies that to obtain $\boldsymbol{G}_0^{(2)}$ we need $\boldsymbol{G}_0^{(3)}$, and so on. So, we have to truncate the QBD after some level M to be able to start the recursion. We may compute $\boldsymbol{G}_M$ with the logarithmic-reduction algorithm as described in [11], that is

$$\boldsymbol{G}_M = \sum_{l=0}^{\infty} \left[\prod_{i=0}^{l-1} U^i_{M-1+2^i}\right] D^l_{M-1+2^l} \tag{13}$$

where the matrices U_k^l and D_k^l are given by Equations (4–7). Accordingly, we obtain the matrices $\boldsymbol{G}_{M-1}$, $\boldsymbol{G}_{M-2}$, $\ldots$, $\boldsymbol{G}_2$ with Equation (10). Using Equation (12), we finally end up with the following system which provides us the absorption probability $\boldsymbol{G}_1$.

$$\boldsymbol{G}_0^{(M)} = Q_3^{(M)} \tag{14}$$

$$\boldsymbol{G}_0^{(M-1)} = \left[I - Q_0^{(M-1)}\, \boldsymbol{G}_M\right]^{-1} \left[Q_3^{(M-1)} + Q_0^{(M-1)}\, \boldsymbol{G}_0^{(M)}\right] \tag{15}$$

$$\vdots$$

$$\boldsymbol{G}_1 = \left[I - Q_0^{(1)}\, \boldsymbol{G}_2\right]^{-1} \left[Q_2^{(1)} + Q_0^{(1)}\, \boldsymbol{G}_0^{(2)}\right] \tag{16}$$

By truncating the QBD at level M, we actually compute the absorbing probability under the taboo of level $M+1$, but a sufficiently large M will provide us a good approximation of absorption probability.

V. Numerical Evaluation

Let us now consider some numerical evaluation of the proposed models, starting with the analysis of the optimistic case. We assume that initially there is a single source sharing both segments in the network, so the system starts at state $(0,0,1)$. The accuracy of our proposed algorithm for computing the extinction probabilities in Section IV-A depends on the term l, at which the infinite sum in Equation (3) is truncated. Experiments show that in our case the accuracy for $l=3$ is already sufficient.

The resulting extinction probability as function over the death rate is illustrated in Figure 3 for file sizes of $F = 9.28\,\mathrm{MB}$ and $F = 6.8\,\mathrm{MB}$, with $Z = 4.64\,\mathrm{MB}$ as defined earlier being the size of the first segment. We can see that when the death rate approaches 1, the extinction probability increases drastically to 1. The smaller file size has the effect that the second segment is transmitted faster and thus more copies of it exist in the network, which reduces the overall extinction probability slightly. In general, this result can be interpreted as follows. The average death rate d corresponds to the reciprocal of the average sharing time of a peer in the system in seconds. Thus, in order for the content provider to keep a low extinction probability of about 0.01, he should provide incentives that peers stay in the system for at least 100 s.

We now look at the more pessimistic case that the dissemination stops when at least one segment is no longer available for sharing. This is shown in Figure 4 for $F = 9.28\,\mathrm{MB}$ and a fixed death rate $d = 10^{-2}$.

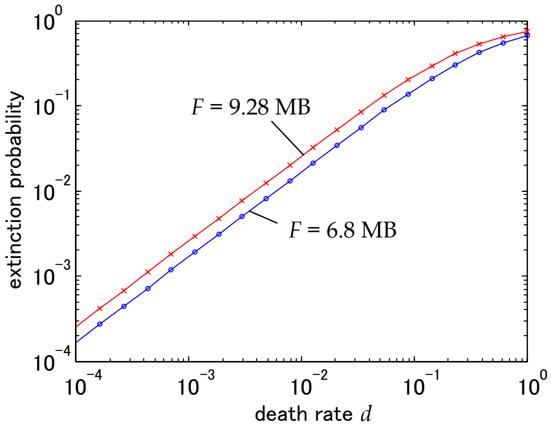

Fig. 3. Extinction probability for different file sizes

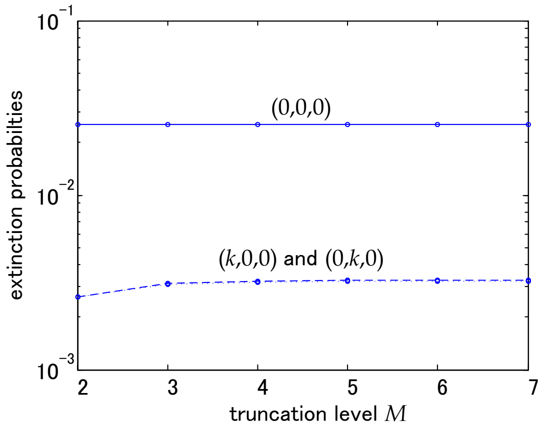

Fig. 4. Influence of the truncation level M on the accuracy

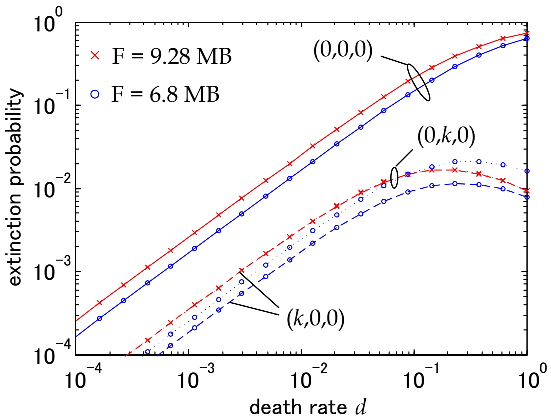

Fig. 5. Extinction probabilities with catastrophes ($M = 5$)

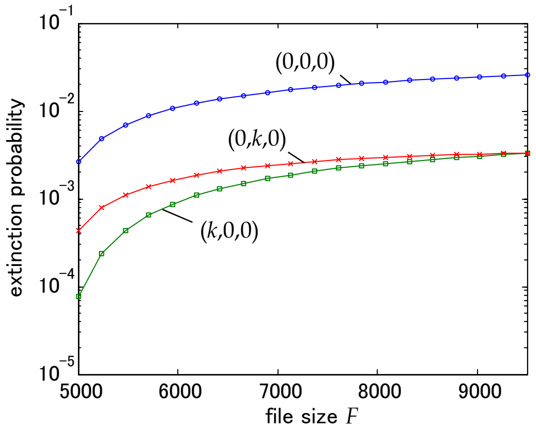

Fig. 6. Influence of file size F on extinction probabilities

For the probability that none of both segments are left in the system, i.e., case $(0, 0, 0)$, we can see that all probabilities are identical and are not affected by the truncation level M. However, a slight difference can be seen when we compare the probabilities where only one kind of segment becomes extinct. In fact, Figure 4 shows that a value of about $M = 5$ proves to be accurate enough, so in the following evaluations we will use this truncation point.

If we show the extinction probabilities from the second model with catastrophes over the death rate, we can recognize in Figure 5 that the probabilities for $(0, 0, 0)$ lie above the two curves $(k, 0, 0)$ and $(0, k, 0)$. The reason why they are larger can be interpreted as follows. Initially, the system starts at state $(0, 0, 1)$, i.e., with exactly a single sharing peer. In order to reach the absorbing state $(0, 0, 0)$, this peer may either make a direct transition by leaving the system or an indirect path, by first giving birth to other peers which then all leave after time. On the other hand, in order to reach one of the other absorbing states $(k, 0, 0)$ or $(0, k, 0)$ at least one birth must take place to increment S_1 or S_2, respectively. Thus, a direct transition from $(0, 0, 1)$ to an absorbing state of that type does not exist in this case causing a reduction in the weight of the probability.

Additionally, when looking at the shape of the curves, we can recognize that both curves for $(k, 0, 0)$ and $(0, k, 0)$ are identical when we consider equal segment sizes and the probability for finding and sharing both segments is equal. When $F = 6.8\,\mathrm{MB}$ the second segment is only half in size of the first, which results in a higher extinction probability of the first segment. The curves lie below the corresponding curves for $F = 9.28\,\mathrm{MB}$ when the death rate d is small. However, in both cases we can see that when the death rate exceeds 10^{-1} the extinction probabilities drop again. At this point it is more likely that the sharing process will stop before any segment is actually downloaded at all, i.e., $d \gg \mu_1(1,1) + \mu_2(1,1)$, where $\mu_1(1,1) + \mu_2(1,1)$ corresponds to the rate of observing a first new peer with any one of the two segments.

The influence of the file size F and, thus, the different size of the second segment is illustrated in Figure 6. We can recognize that for a death rate of $d = 10^{-2}$ the extinction probabilities increase with the file size and that when the second segment is small, the difference between the extinction probabilities of states $(k, 0, 0)$ and $(0, k, 0)$ is large. As expected, when both sizes are equal, both curves approach the same value.

VI. Conclusion

We provided in this work an algorithmically tractable analysis of a level-dependent QBD process with and without catastrophe, in terms of absorbing probability. We showed its applicability to the modeling of file diffusion in unstructured P2P file sharing networks. Numerical results have confirmed that there is a need for the content provider to offer incentives to the peers to encourage sharing and a long sojourn time in the system in order to maintain a sufficiently low extinction probability.

In the future we will use this model to analytically derive further performance measures, especially transient ones such as the distribution of the number of peers present in the system. Furthermore, we would like to enhance the model to consider a more sophisticated peer behavior by including, e.g., their willingness to share, impatient peers, and pollution.

References

[1] N. Bean and G. Latouche. Numerical approximations for QBDs with infinitely many phases. Technical report, Université Libre de Bruxelles, Département d'Informatique, 2007.

[2] G. Carofiglio, R. Gaeta, M. Garetto, P. Giaccone, E. Leonardi, and M. Sereno. A statistical physics approach for modelling P2P systems. In *ACM MAMA'05*, Banff, Canada, June 2005.

[3] N. Christin, A. S. Weigend, and J. Chuang. Content availability, pollution and poisoning in file sharing peer-to-peer networks. In *Proc. of ACM EC'05*, Vancouver, BC, 2005.

[4] Z. Ge, D. Figueiredo, S. Jaiswal, J. Kurose, and D. Towsley. Modeling peer-peer file sharing systems. In *Proc. of IEEE INFOCOM 2003*, pages 2188– 2198, San Francisco, CA, March 2003.

[5] S. Hautphenne, K. Leibnitz, and M.-A. Remiche. Extinction probability in peer-to-peer file diffusion. *SIGMETRICS Perform. Eval. Rev.*, 34(2):3–4, 2006.

[6] T. Hoßfeld, K. Leibnitz, R. Pries, K. Tutschku, P. Tran-Gia, and K. Pawlikowski. Information diffusion in eDonkey-like P2P networks. In *ATNAC 2004*, Bondi Beach, Australia, 2004.

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

[8] K. Leibnitz, T. Hoßfeld, N. Wakamiya, and M. Murata. On pollution in eDonkey-like peer-to-peer file-sharing networks. In *Proc. of GI/ITG MMB 2006*, Nuremberg, Germany, March 2006.

[9] K. Leibnitz, T. Hoßfeld, N. Wakamiya, and M. Murata. Peer-to-peer vs. client/server: Reliability and efficiency of a content distribution service. In *Proc. of ITC-20*, Ottawa, Canada, June 2007.

[10] D. Qiu and R. Srikant. Modeling and performance analysis of BitTorrent-like peer-to-peer networks. In *ACM SIGCOMM'04*, Portland, OR, August 2004.

[11] V. Ramaswami and P. Taylor. Some properties of the rate operators in level dependent quasi-birth-and-death processes with a countable number of phases. *Communications in Statistics: Stochastic Models*, 12(1):143–164, 1996.

[12] D. Rubenstein and S. Sahu. Can unstructured P2P protocols survive flash crowds? *IEEE/ACM Trans. Netw.*, 13(3):501–512, 2005.

[13] S. Saroiu, K. P. Gummadi, R. J. Dunn, S. D. Gribble, and H. M. Levy. An analysis of internet content delivery systems. In *Proc. of USENIX OSDI 2002*, Boston, MA, Dec. 2002.

[14] I. Stoica, R. Morris, D. Karger, M. F. Kaashoek, and H. Balakrishnan. Chord: A Scalable Peer-to-peer Lookup Service for Internet Applications. In *Proc. of ACM SIGCOMM'01*, San Diego, CA, Aug. 2001.

[15] X. Yang and G. de Veciana. Service capacity of peer to peer networks. In *Proc. of IEEE INFOCOM 2004*, pages 2242 – 2252, Hong Kong, March 2004.