Content Availability and Bundling in Swarming Systems

Daniel S. Menasche
University of Massachusetts, Amherst, MA, USA
sadoc@cs.umass.edu

Antonio A. A. Rocha
Federal University of Rio de Janeiro, RJ, Brazil
arocha@land.ufrj.br

Bin Li
Tsinghua University, Beijing, China
libinhere@gmail.com

Don Towsley
University of Massachusetts, Amherst, MA, USA
towsley@cs.umass.edu

Arun Venkataramani
University of Massachusetts, Amherst, MA, USA
arun@cs.umass.edu

ABSTRACT

BitTorrent, the immensely popular file swarming system, suffers a fundamental problem: content unavailability. Although swarming scales well to tolerate flash crowds for popular content, it is less useful for unpopular content as peers arriving after the initial rush find the content unavailable.

Our primary contribution is a model to quantify content availability in swarming systems. We use the model to analyze the availability and the performance implications of bundling, a strategy commonly adopted by many BitTorrent publishers today. We find that even a limited amount of bundling exponentially reduces content unavailability. Surprisingly, for swarms with highly unavailable publishers, the availability gain of bundling can result in a net improvement in download time, i.e., peers obtain more content in less time. We empirically confirm the model's conclusions through experiments on PlanetLab using the mainline BitTorrent client.

Categories and Subject Descriptors

C.4 [**Performance of Systems**]: Modeling techniques

General Terms

Measurement; Performance; Reliability; Theory

1. INTRODUCTION

Despite the tremendous success of BitTorrent (estimated to account for 30–50% of all Internet traffic today), it suffers from a fundamental problem: availability. Although peer-to-peer swarming in BitTorrent scales impressively to tolerate massive flash crowds for popular content, swarming does little to disseminate unpopular content as their availability is limited by the presence of a seed or publisher. The extent of publisher unavailability is severe, e.g., our measurement study shows that 40% of the swarms have no publishers available more than 50% of the time.

To appreciate the availability problem, consider a swarm for an episode of a popular TV show. When a publisher first posts the episode, a flash crowd of peers joins the swarm to download it. The original publisher goes offline at some point, but peers may continue to obtain the content from other peers while the swarm is active. If a peer arrives after the initial popularity wave, when the population of the swarm has dwindled down to near-zero, it finds the content unavailable and must wait until a publisher reappears.

Our primary contribution is a mathematical model to study content availability in swarming systems such as BitTorrent. We use an $M/G/\infty$ queue to model the self-scaling property of BitTorrent swarms, i.e., more peers bring in more capacity to the system. The key insight is to model uninterrupted intervals during which the content is available as *busy periods* of that queue. The busy period increases exponentially with the arrival rate of peers and publishers and the time spent by peers and publishers in the swarm.

Our model enables us to analyze the impact of *bundling*, a common strategy adopted by BitTorrent publishers wherein, instead of disseminating individual files via isolated swarms, a publisher packages a number of related files and disseminates it via a single larger swarm. To appreciate why bundling improves content availability, consider a bundle of K files. Assume that the popularity of the bundle is roughly K times the popularity of an individual file as a peer requesting any file requests the entire bundle. The size of the bundle is roughly K times the size of an individual file. Our model suggests that the busy period of the bundled swarm is a factor $e^{\Theta(K^2)}$ larger than that of an individual swarm. Indeed, if busy periods supported by peers alone last until a publisher reappears, the content will be available throughout.

Surprisingly, in some cases, the improved availability can reduce the download time experienced by peers, i.e., peers download more content in less time. The *download time* of peers in the system consists of the *waiting time* spent while content is unavailable and the *service time* spent in actively downloading content. If the reduction in waiting time due to bundling is greater than the corresponding increase in service time, the download time decreases. We validate this conclusion in Section 4 through large-scale controlled experiments using the mainline BitTorrent client over Planetlab.

CoNEXT'09, December 1–4, 2009, Rome, Italy.

Our experiments also show that the conclusions of our model qualitatively hold even with realistic arrival patterns, peer upload capacities, and heterogeneous popularities.

In summary, we make the following contributions.

Measurement: We present a large-scale measurement study of real BitTorrent swarms that shows that (1) content availability is a serious problem due to publisher unavailability, (2) bundling of content is a widely prevalent, and (3) bundled content is more available than unbundled content.

Availability model: We present a novel queuing-theoretic model to analyze content availability and download times in BitTorrent-like swarming systems. To our knowledge, this is the first model that relates content availability to arrivals and departures of peers as well as publishers.

Implications of bundling: We use the model to analyze the implications of bundling, a widely prevalent yet little studied phenomenon, and show that (1) bundling improves availability, and (2) bundling can reduce download times for unpopular content when publishers are highly unavailable.

Experimental validation: We validate the model using large-scale controlled experiments with the mainline BitTorrent client on PlanetLab showing that the model accurately predicts download times in swarms with intermittently available publishers for both bundled and individual content.

2. MEASURING CONTENT AVAILABILITY AND BUNDLING IN BITTORRENT

In this section, we present a large-scale measurement study of BitTorrent that shows that 1) content unavailability is a serious problem in BitTorrent today, and 2) bundling of content is widely prevalent and bundled contents shows greater availability. We begin with a brief overview of how swarming in BitTorrent works and why content becomes unavailable.

2.1 Why unavailability?

A swarm consists of a set of peers concurrently sharing (downloading or uploading) content (a file or a bundle of files) of common interest with the help of a coordinating tracker. Content is divided into blocks and peers obtain meta-data about constituent blocks as well as identities of other peers in the swarm from the tracker. A peer exchanges blocks with other peers using a tit-for-tat incentive strategy until it completes its download. Peers that have not yet completed their downloads are called *leechers* while peers that possess all blocks in the content are called *seeds*.

Content is available if either at least one seed is present or sufficiently many active leechers are present so as to collectively make all constituent blocks available. Seeds may become unavailable in practice due to several reasons. Publishing sites serving a large number of files may take down seeds after the initial popularity wave subsides in order to reduce bandwidth costs. A seed may also be an average user publishing home-generated content that can not afford to stay online all the time. Seeds illegally uploading copyrighted material often disappear quickly for obvious reasons. Even for legitimate content, maintaining highly available seeds entails administrative effort and cost, which runs counter to the goals of content publishers that value BitTorrent as a cheap alternative to a client-server approach.

Throughout this section, we measure content availability by equating it with seed availability. In the next section, we model content availability resulting both from seeds as well as from leechers alone. In the rest of this paper, we use the terms *publishers* and *peers* interchangeably with *seeds* and *leechers* respectively.

2.2 Measuring unavailability

How available is content in BitTorrent swarms today? To answer this question, we conducted a seven-month long measurement study of BitTorrent swarms as follows. We developed and deployed BitTorrent monitoring agents at 300 nodes on Planetlab from August 3, 2008 to March 6, 2009. Once every hour, a host at the University of Massachusetts Amherst receives an RSS feed advertised by GoogleReader of recently created torrent URLs from Mininova (a large torrent hosting site), and sends each URL to a subset of the monitoring agents on Planetlab. The agents fetch the torrent metadata by joining the swarm and begin to monitor its peers. Our agents leverage the Peer Exchange (PEX) protocol extension, that enables it to discover new neighbors from other peers in addition to the tracker. To avoid copyright issues, our agents collect information only about the control plane without actually uploading or downloading content, which suffices for our purposes as we equate content availability with seed availability.

To distinguish seeds from leechers, our agents record the bitmaps received from connected peers. The bitmaps are part of the BitTorrent protocol and a peer uses them to convey the blocks it possesses to its neighbors. Each entry in the trace collected by the agents consists of a swarm identifier, a peer identifier (IP address and port number) and its bitmap recorded roughly periodically for each discovered peer in the swarm. Our traces consist of more than 14 million distinct IP addresses and 66K distinct swarms.

Figure 1 shows the distribution of seed availability for the monitored swarms. The solid curve shows the availability in the first month after the creation of the swarm, when we expect the content to be more popular. The extent of publisher unavailability is severe: less than 35% of the swarms had at least one seed available all the time. The availability of swarms over the entire duration of the measurement is even lower as shown by the dotted curve: almost 80% of the swarms are unavailable 80% of the time.

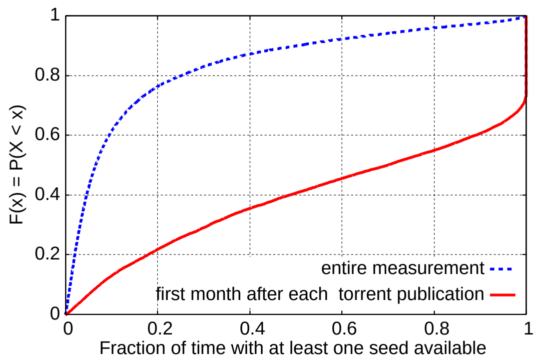

Figure 1: CDF of seed availability in 45,693 swarms each monitored for at least one month.

2.3 Content bundling

Bundling of content is a common practice in BitTorrent today. In this section, we study the extent of bundling and its impact on availability. The trace used in this section is a snapshot of BitTorrent swarms taken on May 6, 2009. For each of the 1,087,933 swarms in this snapshot, we record its content category (e.g., movies, TV, books etc.), names and sizes of constituent files, creation date, and instantaneous number of seeds and leechers. Note that we could afford to monitor many more swarms in this dataset than the previous one as we did not have to measure details of peer arrivals and departures inside each swarm.

2.3.1 Extent of bundling

We analyze the extent of bundling in three of the nine categories present in Mininova, namely, music, TV shows and books. These three categories together account for 45.98% of the swarms and 31.93% of the peers in the system. We chose these three categories because it is easier to automatically detect bundling by checking for the presence of multiple files with known extensions (e.g., `.mp3` for songs, `.mpg` for TV shows and `.pdf` for books). Detecting bundling is non-trivial in some categories, e.g., a DVD for a single movie is often organized as a collection of video files that are never distributed individually, making it difficult to check for the presence of multiple movies without manual inspection.

Among music swarms, albums are common. We classify a music swarm as a bundle if it has two or more files with common audio file extensions such as `.mp3`, `.mid` and `.wav`, which results in 193,491 of the 267,117 monitored swarms being classified as bundles.

Among TV show swarms, many bundles consist of sets of episodes in a season. We classify swarms that have two or more files with common video file extensions such as `.mpg` and `.avi` as bundles, which results in 25,990 of the 164,930 monitored swarms being classified as bundles.

Among book swarms, we observe that collections, i.e., torrents containing the keyword "collection" in their titles, usually consist of a bundle of contents connected by a broad theme, e.g., the "Ultimate Math Collection (1)" of size 5.81 GB has 642 books. We classified 841 of the 66,387 monitored swarms as collections. Classifying swarms that contain 2 or more files with common file extensions such as `.pdf` and `.djvu` as bundles results in an additional 6,270 bundles.

2.3.2 Bundled content is more available

In this section, we present evidence suggesting that bundling is correlated with higher availability. We first consider book swarms. We find that 62% of all book swarms had no seed available on May 6, 2009, whereas that number drops to 36% if we consider only collections. Furthermore, the average number of downloads for a typical book swarm is 2,578, whereas for collections it is 4,216.

One reason for higher seed availability may be that content publishers are intrinsically more willing to support seeds for bundled content. The higher number of downloads for bundled content may be either because of higher demand for bundled content (as any peer seeking any of the constituent files may opt to download the bundle), or because of higher availability, or both. Higher seed availability in turn may in part be because of the increased number of downloads as some peers may choose to altruistically disseminate the content further. Although it is difficult to discern cause and effect in our measurement data, our analytic model in the next section quantifies how the higher demand and higher seed availability for bundled content *produce* improved content availability.

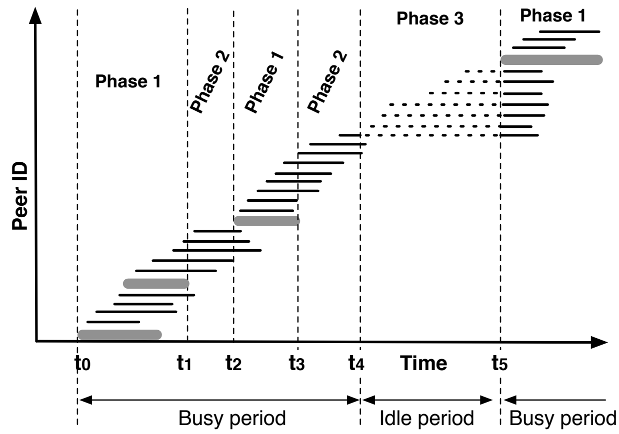

Figure 2: Illustration of busy and idle periods.

We next analyze our traces more closely for content that is available both in isolation and as part of a larger bundle. We observe that among the unavailable collections, some of them were subsets of bigger collections, e.g., the 23 swarms consisting of collections of Garfield comics from 1978 to 2000 had no seeds. However, each of these collections can be found in a single super-collection aggregating all Garfield comics. The super-collection had seven seeds. After a manual inspection of all 841 book collections, we concluded that 210 had no seeds and were not subsets of other collections, which results in 210/841 = 25% unavailability for content disseminated through collections (compared to 62% above for a typical swarm).

As another example, we consider swarms for the popular TV show "Friends". There were a total of 52 swarms associated with this show. Among them, 23 had one or more seeds available, and the remaining 29 had no seeds. The 23 available swarms consisted of 21 bundles (and 2 single episodes), whereas the 29 unavailable swarms consisted of only 7 bundles. These observations suggest a strong correlation between bundling and higher availability. The next section presents an analytic model that quantifies the causal relationship between the two.

3. MODEL

In this section, we develop a model for content availability in BitTorrent. The key insight underlying the model is to view BitTorrent as a *coverage process* or equivalently an $M/G/\infty$ queuing system. The model shows that 1) bundling improves availability, and 2) for swarms with highly unavailable publishers, the availability benefit of bundling more than offsets the increased time to actively download more content, resulting in a net decrease in user-perceived download times.

3.1 Model overview

Figure 2 illustrates how content availability in BitTorrent depends upon the arrivals and departures of publishers and peers. Each horizontal line segment represents the time interval during which a peer (represented using thin lines) or a publisher (represented using thick lines) stays online. A swarm is initiated by the arrival of a publisher, which also marks the start of the first *busy period*. The swarm's lifetime is divided into alternating busy and *idle periods*. Content is available during busy periods and unavailable during idle periods. If a publisher is always online, the first busy period lasts forever and content remains always available.

A busy period ends when the following two conditions are satisfied: 1) there are no publishers online, and 2) the *coverage*, i.e., the number of peers currently online, drops below a fixed small threshold (causing some blocks to become unavailable). For example, Figure 2 shows that after all publishers leave at time t_1, the busy period continues with the help of peers alone until a publisher reappears at time t_2. A busy period may alternate any number of times between a phase consisting of one or more publishers (Phase 1) and a phase consisting of peers alone (Phase 2). Peers arriving during either phase in a busy period will find the content available. At t_4, there are no publishers and the number of peers drops below the coverage threshold (assumed 3 in this example). This initiates an idle period that lasts until a publisher reappears at time t_5. Extant peers at the end of a busy period as well as peers arriving during the idle period find the content unavailable (represented by dotted lines). Because of idle waiting, these peers experience longer *download times* defined as the times since a peer arrives until it completes the download.

Our goal is to understand how content availability and the download times experienced by peers in a swarm depend upon 1) its popularity or the peer arrival rate λ; 2) the mean time s/μ that a peer takes during a busy period to actively download the content of size s at a rate equal to the effective average capacity μ of the swarm; and 3) the arrival rate r of publishers and the mean time u that a publisher stays online. We have implicitly assumed that u must be long enough for at least one copy of the file to be served in each busy period. For simplicity, we have assumed that peers are selfish and leave as soon as they complete their download; §3.3.4 extends the model to incorporate altruistic lingering.

To appreciate why bundling improves content availability, consider the special case of a highly unavailable publisher, i.e., its arrival rate r and mean residence time u are small. Then, the length of a busy period is determined primarily by peer arrivals and departures. Assuming Poisson peer arrivals and a coverage threshold of one, the length of a busy period can be shown to be $\frac{e^{\lambda s/\mu}-1}{\lambda}$. Bundling K files increases the peer arrival rate for the bundle to $K\lambda$ as each peer desiring any of the constituent files requests the entire bundle, and increases the time spent by each peer in the swarm to Ks/μ. As a result, the length of the busy period for the bundled swarm is $\frac{e^{K^2\lambda s/\mu}-1}{K\lambda}$, which translates to a reduction in unavailability by a factor $e^{\Theta(K^2)}$. For highly unavailable publishers, the availability gains of bundling can outweigh the cost of the increased time to download K times as much content resulting in a reduction in the download time, i.e., peers obtain more content in less time.

The rest of this section formalizes the above claims and derives closed-form expressions for the total download time experienced by peers with and without bundling. Unless otherwise stated, we assume that inter-arrival times of peers and publishers, residence time of publishers, and file download times are all exponentially distributed.

Variable	Description (units)
λ_k	peer arrival rate (1/s)
$\Lambda = \sum_{i=1}^{K} \lambda_k$	bundled peer arrival rate (1/s)
s_k	file size (bits)
$S = \sum_{i=1}^{K} s_k$	bundle size (bits)
μ	mean download rate of peers (bits/s)
r_k	arrival rate of publishers (1/s)
R	arrival rate of publishers for the bundle (1/s)
u_k	mean publisher residence time (s)
U	mean bundled publisher residence time (s)
Metric	**Description (units)**
P_k	unavailability
$\mathcal{P}$	unavailability of bundle
T_k	download time (s)
$\mathcal{T}$	bundle download time (s)

Table 1: Variables denoted by lower case characterize swarm $k \in \{1, 2, \cdots, K\}$ in isolation, while variables denoted by capital letters characterize the bundle of K files. Metrics for the swarms in isolation and for bundles are denoted by plain and stylized letters, respectively. Subscripts are dropped when homogeneous files are considered.

3.2 A simple model for content availability

We present a simple instance of the above model to analyze content availability and show that bundling improves availability. The model makes several simplifying assumptions (which we progressively relax in subsequent sections), but brings out the key insight underlying all of our results.

Assumptions: Content is available if and only if there is at least one publisher online. A peer arriving during an idle period finds the file unavailable and immediately leaves, i.e., it does not queue up until a publisher arrives. ⋄

Availability of an individual swarm.

In swarm k, let r_k and u_k denote the arrival rate and residence time of publishers (refer to Table 1 for notation). Swarm k cycles through busy and idle periods, with average length $E[B_k]$ and $1/r_k$, respectively. The probability P_k that a peer arrives to swarm k to find the content unavailable is

$$P_k = \frac{1/r_k}{E[B_k] + 1/r_k}, \qquad k = 1, \ldots, K \tag{1}$$

and

$$E[B_k] = \frac{e^{r_k u_k} - 1}{r_k} \tag{2}$$

The above follows from classical results for the busy period of an M/G/∞ queue.

Availability of a bundled swarm.

Let R and U denote the arrival rate and residence time of publishers for the bundle, respectively. The probability $\mathcal{P}$ that a peer arrives to find the content unavailable in the bundled swarm is

$$\mathcal{P} = \frac{1/R}{E[\mathcal{B}] + 1/R} \tag{3}$$

where the average length of a busy period for a bundle of K files is

$$E[\mathcal{B}] = \frac{e^{RU} - 1}{R} \qquad (4)$$

Consider the special case when the publisher arrival rates are the same for all files, i.e., $r_k = r$ and their residence times are also the same, i.e., $u_k = u$ for all K files. If R and U scale as $R = Kr$ and $U = Ku$, then

$$E[\mathcal{B}] = \frac{e^{K^2 ru} - 1}{Kr} \qquad (5)$$

$$\mathcal{P} = e^{-K^2 ru} \qquad (6)$$

Note that $E[\mathcal{B}]$ is a factor $e^{\Theta(K^2)}$ larger than the corresponding value for an individual swarm. It is straightforward to see that $-\log P_k = \Theta(1)$ and $-\log \mathcal{P} = \Theta(K^2)$. Thus, bundling reduces content unavailability by a factor $e^{-\Theta(K^2)}$.

Availability with publishers and peers.

Assumptions: The busy period is defined w.r.t. a coverage threshold of one, i.e., a peer arriving during a busy period always finishes the download in that busy period and the last peer to finish ends the busy period. ⋄

Content may be available even if there are no publishers online. Let the aggregate arrival rate of peers and publishers to the individual swarm and to the bundle be $\lambda_k + r_k$ and $\Lambda + R$, respectively. For simplicity, we first consider the special scenario in which publishers stay for a time equal to that of disseminating one copy of the file, i.e., $u_k = s_k/\mu$ (an assumption we relax in the following section),

$$E[B_k] = \frac{e^{(r_k+\lambda_k)s_k/\mu} - 1}{r_k + \lambda_k}, \qquad k = 1, \ldots, K \qquad (7)$$

and

$$E[\mathcal{B}] = \frac{e^{(R+\Lambda)S/\mu} - 1}{\Lambda + R} \qquad (8)$$

If for all K files, $\lambda_k = \lambda$ and $s_k = s$, i.e., $\Lambda = K\lambda$ and $S = Ks$, then the bundled busy period is $E[\mathcal{B}] = e^{\Theta(K^2)}$. Thus, bundling reduces the unavailability by $e^{-\Theta(K^2)}$ (for any fixed arrival rate R of the bundled publisher).

3.3 A model for content availability and download time

Next, we quantify content availability and the mean download time experienced by peers when 1) peers may wait for content to become available, 2) the mean residence time of the publisher may differ from the service time of peers and 3) the coverage threshold may be greater than one.

We begin by presenting the theoretical background required by our model. Our results rely on those reported by Browne and Steele [2] on the busy period of an M/G/∞ queue where the customer initiating the busy period has an exceptional residence time.

Let customers arrive according to a Poisson process with rate β. The residence time of the customer initiating a busy period is draw from an exponential distribution with mean θ. The residence time of all other customers, X, takes the form of one of two exponentially distributed random variables, X_1 or X_2, with averages α_1 and α_2, respectively; $X = X_1$ with probability q_1 and $X = X_2$ with probability $q_2 = 1 - q_1$. The expected busy period is

$$E[B] = \theta + \sum_{i=1}^{\infty} \frac{\beta^i}{i!} \sum_{j=0}^{i} \binom{i}{j} \frac{q_1^j q_2^{i-j} \alpha_1^{1+j} \alpha_2^{1-j+i} \theta}{\alpha_1\alpha_2 + j\theta\alpha_2 + \theta\alpha_1 i - \theta\alpha_1 j} \qquad (9)$$

The reader can find the derivation of (9) in the Appendix. The proofs of the results that follow, when not included in the Appendix, are in the technical report [10]. In the rest of this section, unless otherwise stated, we assume that all files have the same size and demand, and that the publisher arrival rates and residence times are the same across all swarms. Assuming homogeneous swarms allows us to drop the subscripts of variables referring to individual swarms. In [10] we show that most of our results extend to the case where different swarms have different characteristics.

3.3.1 *Availability with impatient peers*

Assumptions: Publishers arrive to individual swarms at rate r and stay in the system for a mean time u. For the bundled swarm, publishers arrive with rate R and stay for a mean time U. Peers that arrive during an idle period leave immediately. ⋄

We are interested in determining the probability that a request leaves without being served. Denote this as P and $\mathcal{P}$ for the individual and bundled systems, respectively. Then

$$P = \frac{1/r}{E[B] + 1/r} \qquad \mathcal{P} = \frac{1/R}{E[\mathcal{B}] + 1/R} \qquad (10)$$

The average busy period for each individual swarm, $E[B]$, is obtained from (9) by setting the parameters as follows: $\beta = \lambda + r$, $\theta = u$, $\alpha_1 = s/\mu$, $q_1 = \lambda/(\lambda + r)$, $\alpha_2 = u$.

For the bundled swarm, the aggregate peer arrival rate is $\Lambda = K\lambda$ and the size is $S = Ks$. The average busy period, $E[\mathcal{B}]$, is obtained from (9) as follows: $\beta = \Lambda + R$, $\theta = U$, $\alpha_1 = S/\mu$, $q_1 = \Lambda/(\Lambda + R)$, $\alpha_2 = U$.

The following lemma concerns the number of peers served in a busy period. Assuming that both the bundle publisher arrival rate, R, and publisher residence time, U, are independent of K, we have

LEMMA 3.1. *The mean number of peers served in a busy period, $E[\mathcal{N}]$, increases as $e^{\Theta(K^2)}$ by bundling K files.*

Note that this result is qualitatively similar to the case when publishers and peers stay online for the same mean time (Section 3.2).

We now consider the scenario where peers have skewed preferences. Given K contents, let p_k denote the probability that a request is for content k, $k = 1, \ldots, K$. Assume that $p_k = c/k^\delta$, $\delta > 0$ (Zipf's law). Letting Λ denote the aggregate peer arrival rate for all K swarms, the arrival rate for swarm k is $\lambda_k = p_k\Lambda$. Under the assumption that the mean time to download the bundle scales as K/μ, one can show that the lemma above still holds (details in [10]).

In the theorem below we relate the asymptotics of the busy period to the probability that a request is not served. Under the same assumptions of Lemma 3.1 we have,

THEOREM 3.1. **(Availability theorem)** *Bundling K files together decreases unavailability by a factor $e^{\Theta(K^2)}$.*

Although the above result assume the publisher arrival rate for the bundle, R, to be constant and independent of

K, we show in [10] that even if $R = \Omega(e^{-cK^2})$, $c > 0$, the availability of the bundle is still greater than the availability of the individual swarm by a factor $e^{\Theta(K^2)}$. When enough files are bundled, the long busy periods of the bundled swarm make it nearly self-sustaining, so peers can almost always download the content even in the absence of publishers.

3.3.2 Mean download time with patient peers

Assumptions: Peers that arrive during an idle period wait for a publisher to become available. The other assumptions are the same as in Section 3.3.1. ⋄

We wish to compare the download time of peers with and without bundling. To this aim, we first compute the average busy period length in an individual swarm, $E[B]$. When content is unavailable and a publisher arrives to start a busy period, the group of waiting peers immediately begins to be served. Neglecting the possible impact of this group of peers on the duration of the busy period, the average busy period $E[B]$ can be obtained from (9) by setting $\beta = \lambda + r$, $\alpha_1 = s/\mu$, $q_1 = \lambda/(\lambda + r)$, $\alpha_2 = \theta = u$. In the technical report [10] we also provide an expression for $E[B]$ accounting for the possible impact of the group of peers that begins to be served when the publisher arrives.

The mean download time, $E[T]$, is given by

Lemma 3.2. *The mean download time of a file when peers are patient is*

$$E[T] = \frac{s}{\mu} + \frac{1}{r}P \qquad (11)$$

where $P = \frac{1/r}{1/r+E[B]}$.

For the bundled swarm, the mean busy period length, $E[\mathcal{B}]$, can be obtained from (9) by setting $\beta = \Lambda + R$, $\alpha_1 = S/\mu$, $q_1 = \Lambda/(\Lambda + R)$, $\alpha_2 = \theta = U$. Once $E[\mathcal{B}]$ is obtained, the mean download time for the bundle, $E[\mathcal{T}]$, can be derived from (11) replacing s, r and $E[B]$ by their bundle counterparts S, R and $E[\mathcal{B}]$.

In the following theorem we relate the mean download times of bundles and individual swarms,

Theorem 3.2. **(Download time theorem)** *Bundling K files,*

(a) can increase the mean download time of each file by at most a factor K;

(b) can decrease the mean download time of each file by a factor $\Theta(1/R)$ which grows unbounded as $R \to 0$.

When service times dominate download times, bundling can increase the download time by up to a factor of K as peers download K times as much content. Nevertheless, when wait times dominate download times as is the case with highly unavailable publishers, peers may experience arbitrarily smaller download times when downloading bundles.

3.3.3 Threshold coverage

Assumptions: Same as those described in §3.1. ⋄

If a peer leaves the system carrying the last copy of a chunk, content may become unavailable even if the number of peers online, i.e., the coverage, is greater than one. Our aim now is to determine the availability and the mean download time experienced by peers in the general case where content becomes unavailable when no publisher is online and the coverage reaches a threshold m.

Let $B(n, m)$ be the expected length of a *residual busy period* that begins with n leechers and ends as soon as the population size reaches m. The mean busy period corresponds to $B(1, 0)$. $B(n, m)$ is given by

Lemma 3.3. *For all n,*

$$B(n, 0) = \sum_{i=1}^{n} \frac{s}{i\mu} + \frac{s}{\mu} \sum_{i=1}^{\infty} \left(\frac{s\lambda}{\mu}\right)^i \frac{(n+i)! - n!i!}{i!(n+i)!i} \qquad (12)$$

For $m < n$, $B(n,m)$ is obtained using the recursion $B(n, m) = B(n, 0) - B(m, 0)$.

We use Lemma 3.3 to estimate the unavailability probability and the expected download time of peers in the scenario described in §3.1 and depicted in Figure 2. We assume that 1) the distribution of the residual busy period is concentrated around its mean and 2) publishers stay long enough in the system so that, when Phase 2 begins, the population of peers is in steady state. We denote the mean residual busy period starting when the system transitions to Phase 2 by $B(m)$,

$$B(m) = \sum_{i=0}^{\infty} \frac{e^{-\frac{\lambda s}{\mu}} \left(\frac{\lambda s}{\mu}\right)^i}{i!} B(i, m) \qquad (13)$$

Noting that the number of times that the system cycles through Phases 1 and 2 before transitioning to Phase 3 is described by a geometric random variable with mean $e^{rB(m)}$ yields

Theorem 3.3. *For a threshold coverage of m, the mean download time of a file when peers are patient is $s/\mu + P/r$ where*

$$P = \exp(-r(u + B(m))) \qquad (14)$$

The corresponding expression for bundled swarms is obtained by replacing s, λ, r and u by their bundled counterparts, S, Λ, R and U. In particular, if $R = Kr$ and $U = Ku$ the availability and download time theorems still hold. In Section 4.3.1, we validate the mean download time estimated using the above theorem against experiments.

3.3.4 Altruistic lingering

Assumptions: Peers remain in the system for an average amount of time $1/\gamma$ after completing their downloads. The other assumptions are the same as in Section 3.3.2. ⋄

Peers may stay online as seeds after completing their downloads, either because they are altruistic or because publishers provide them incentives to do so. In the technical report [10] we show how to parameterize a general version of equation (9) to derive the availability probability and the mean download time of peers that stay online as seeds after completing their downloads. Furthermore, we show that the availability and the download time theorems still hold.

To illustrate the consequences of peers staying longer in the system, consider two swarms with file sizes s_1 and s_2 and popularities λ_1 and λ_2. We wish to compare the performance of the individual swarms with that of a bundle with similar availability $[s_1\lambda_1/\mu + \lambda_1/\gamma = (\lambda_1 + \lambda_2)(s_1 + s_2)/\mu]$. The mean residence time for requestors of content 1 is equal to

$$\frac{s_1}{\mu} + \frac{1}{\gamma} = \frac{(\lambda_1 + \lambda_2)(s_1 + s_2)}{\mu\lambda_1} = \frac{s_1 + s_2}{\mu}\left(1 + \frac{\lambda_2}{\lambda_1}\right) \qquad (15)$$

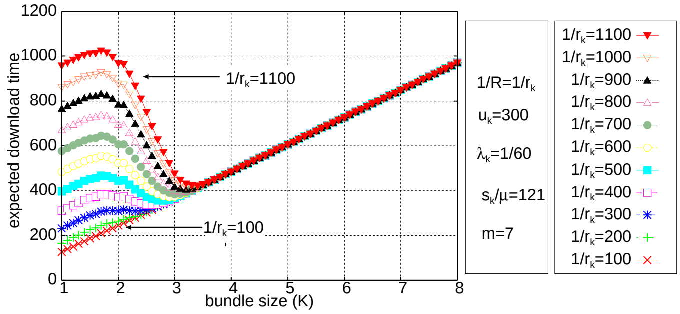

Figure 3: Bundles may reduce download time.

For the bundled swarm, the mean download time of peers is given by $(s_1 + s_2)/\mu$.

Assume swarm 1 is associated with a small and unpopular content while the swarm 2 content is large and popular, $s_1 \ll s_2$, $\lambda_1 \ll 1 \ll \lambda_2$. Since content 1 is very unpopular (peer interarrival time very large), high availability depends on peers staying for a long time in the system after concluding their downloads (in equation (15), $1 + \lambda_2/\lambda_1 \to \infty$ as $\lambda_1 \to 0$). If swarm 1 is bundled with swarm 2, on the other hand, the overhead incurred by the peers only interested in content 2 is marginal (since $s_1 \ll s_2$) but the gains for peers interested in content 1 is remarkable, since requestors for content 1 experience the same availability and performance as those requesting file 2.

3.4 When can bundling reduce download time?

In this section we use the proposed model to illustrate when bundling reduces mean download time. We numerically evaluate equations (11) and (9) by setting the parameters as described in the legend of Figure 3. Figure 3 shows the expected download time as a function of the bundle size. For seven of the scenarios ($1/R \in [500 - 1100]$), increasing K to its optimal value, $K = 3$, leads to a decrease in the expected download time, while setting $K = 1$ is the best strategy for the remaining four. In each curve, as K increases the mean download time first increases, then decreases and finally increases again. The initial performance degradation occurs because small bundles may increase service times without sufficiently increasing the busy period. Figure 3 also shows that the benefits of bundling increase as the value of R decreases.

4. EXPERIMENTAL EVALUATION

In this section, we report on controlled experiments using real BitTorrent clients to validate the two main conclusions of our model: 1) bundling improves availability, and 2) bundling can reduce download times when publishers are highly unavailable. We use an instrumented version of the mainline BitTorrent client [8] and experiment with private torrents deployed on Planetlab. Our experimental setup thus emulates realistic wide-area network conditions, client implementation artifacts, and the impact of realistic upload capacity distributions and arrival patterns that are difficult to capture in an analytic model.

4.1 Experimental setup

Our experiments were conducted using approximately 150 Planetlab hosts and two hosts at the University of Massachusetts at Amherst one of which is designated as the controller of the experiment and another as a Bittorrent tracker. The controller causes peer arrivals, publisher arrivals, and publisher departures by dispatching via `ssh` a command to start or stop the BitTorrent client on a randomly chosen unused Planetlab host. At the end of the experiment, the controller collects the remote traces logged by the instrumented BitTorrent clients. Each client's trace logs the instantaneous download and upload rates every second as well as the fraction of the file downloaded up to that time.

Experimental parameters. Our experiments consist of torrents that publish either a single file of size $S = 4$ MB or a bundle of K files of aggregate size KS. The peer arrival rate for a bundle is assumed to be the sum of the arrival rates of its constituent files. The uplink capacity of each peer is $\mu = 33$ KBps ($\mu = 50$ KBps in §4.3). The publisher's upload capacity is 50KBps for individual as well as bundled torrents. There is only one publisher that alternates between being on and off. The peer arrival rate λ and on/off behavior of the publisher are varied according to the experimental goals as described below.

4.2 Bundling improves availability

Our model suggests that bundling increases availability by increasing the length of busy periods and thereby reducing the reliance on a stable publisher. As an extreme case, we consider a publisher that initiates a swarm and then goes offline never to come back, and look at how long the swarm remains available after the publisher goes offline. We ensure that the publisher stays online long enough for at least one peer to fully download the file. Each peer leaves the system immediately after downloading the file.

We set λ=1/150 peers/second for each file and all other parameters to their default values, and study how the availability of the publisher-less swarm varies with the level of bundling K.

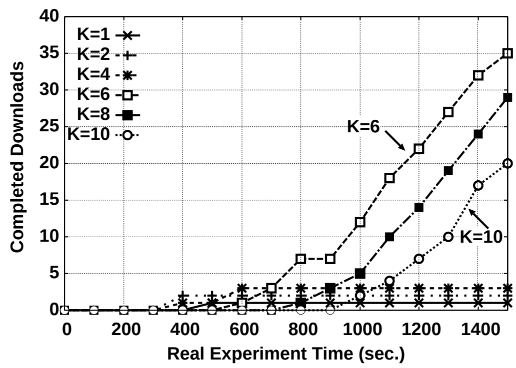

Figure 4: Availability of seedless swarms and the tradeoff in the choice of the bundle size.

Figure 4 shows the number of peers served between 0 and 1500 seconds of the experiment for K=1, 2, 4, 6, 8 and 10. No peer completes its download in the first 300 seconds of the experiment: the publisher is either waiting for the first peer to arrive or is serving the first peer in each case. However, when the first peer completes its download and the publisher goes offline, the curves for $K = 1, 2, 4$ exhibit a very different trend compared to $K = 6, 8, 10$. For $K = 1, 2, 4$, only a small number of additional peers are able to complete their download before parts of the content become unavailable. On the other hand, for $K = 6, 8, 10$, the number of completed downloads increases linearly, i.e., the swarm is self-sustaining even in the absence of a publisher.

In steady state, the length of time the swarm remains self-sustaining after the publisher goes offline is given by the mean residual busy period, $B(m)$. To compute $B(m)$ we use eq. (13) with $\mu = 33$KBps, $s = 4$MB and $\lambda = 1/150$ peers/s. A threshold coverage of $m = 9$ leads to the following values of $B(m)$ for $K = 1$ to 8, (0, 0, 47, 569, 2816, 8835, 256446, 75276). These values capture the fact that for $K \geq 5$ the swarms remained self-sustaining throughout our measurement.

Although the system goes from being unavailable to being available as K increases from 4 to 6, further increasing K only results in increased download times. The average download time of peers when $K = 10$ is roughly 66% higher than that for $K = 6$ (not shown in Figure 4). This suggests a delicate tradeoff in choosing K—it should be large enough to bridge gaps in publisher unavailability, but beyond that point bundling only increases download times. We study this tradeoff in more detail next.

4.3 Bundling can improve download time

In this section, we consider an intermittently available publisher with capacity 100KBps that alternately remains on and off for (exponentially distributed) mean times of 300s and 900s respectively. The arrival rate of peers for each file is $\lambda = 1/60$ peers/second and the capacity of each peer is μ=50 KBps. We study how the average download time of peers varies with the level of bundling.

Figures 5(a)–(c) show peer arrivals and departures over time. Each line segment starts at the instant that the peer arrives and terminates when the peer departs. For each value of K the experiment lasts for 10 runs of 1200s each. Figure 5(a) shows that for $K = 2$, many peers complete their downloads at roughly the same time. These flash departures indicate that the swarm is not self-sustaining. They happen because extant as well as newly arriving peers get stuck soon after the publisher goes off, and must wait until the publisher reappears and serves the missing blocks allowing them to complete their downloads. On the other hand, setting K=3 (Figure 5(b)) reduces the likelihood of peers being blocked, and setting $K = 4$ (Figure 5(c)) nearly eliminates blocking as the swarm sustains itself during periods of publisher unavailability.

Figure 6(a) shows the mean download time as function of K. For $K = 1$ and 2, the mean download time remains large as it is dominated by the time peers spend waiting for the publisher. The large variance is due to the variance in the downtime of the publisher. When $K = 3$, the mean download time reduces significantly, however the variance remains large as the download times are still partly determined by peers waiting for the publisher to reappear. The optimal bundle size is $K = 4$. The mean and the median download time as well as the variance are the lowest for this value of K as bundling eliminates gaps in publisher availability. For values of $K > 4$ the download time increases linearly with respect to K as the download time is dominated by the time to actively download increasingly bigger bundles. The variance continues to remain low as the swarm is increasingly self-sustaining with increasing K.

4.3.1 *Evaluation of the analytical model*

Next, we validate our analytical model (Section 3.3.3) against the experimental results above. We compute the mean download time using Theorem 3.3, adapting (14) to account for the fact that there is only one publisher in the system to obtain

$$\mathcal{P} = \frac{\exp\left(-R\sum_{i=0}^{\infty} \frac{\exp(-\frac{K^2\lambda s}{\mu})(\frac{K^2\lambda s}{\mu})^i}{i!} B(i,m)\right)}{UR+1} \quad (16)$$

The derivation of the formula is in [10]. Setting $s/\mu = 80$s, $\lambda = 1/60$ peers/s, $1/r = 900$ arrivals/s, $u = 300$s and $m = 9$, our model predicts the results observed in Figure 6(a) pretty well. The model leads to an optimal bundle size of $K = 5$, whereas the optimal observed in the experiments was $K = 4$, and correctly captures the trend of the download time curve.

4.3.2 *Heterogeneous upload rates*

Next, we repeat the above experiment with heterogeneous peer upload capacities. The upload rate distribution was taken from the measured data used to generate Figure 1 in the BitTyrant study [14]. The average upload rate is 280KBps and the median is 50KBps. Using realistic peer upload capacities does not qualitatively change the behavior of the system (compare Figures 6(a) and Figures 6(b)). However, the optimal bundle size is now $K = 5$. This is consistent with the increase in the average upload capacity compared to the values obtained from the experiments with homogeneous capacities (μ=50 KBps). The larger upload capacity implies that a larger bundle is needed to increase the length of its busy periods so as to make the swarm self-sustaining during periods of publisher unavailability—a conclusion that agrees with our model.

4.3.3 *Heterogeneous file popularites*

Next, we study the impact of bundling when different files have different popularities. We consider a bundle of four files. We assume that the popularities of the files inside the bundle are distributed as follows: $\lambda_1 = 1/8$, $\lambda_2 = 1/16$, $\lambda_3 = 1/24$ and $\lambda_4 = 1/32$. We run 5 experiments, the first four corresponding to swarms with individual files (experiments 1, 2, 3 and 4) and the last one to a bundle of all the files (experiment 5). In experiment i ($1 \leq i \leq 4$) we set λ_i as described above, and in experiment 5 we set $\lambda = \sum_{i=1}^{4} \lambda_i = 1/3.84$. All other parameters are set to their default values.

The mean download times are illustrated in Figure 6(c). The boxplots and lines show the distribution quartiles and 5th and 95th percentiles. For the individual files, as we move to the right in Figure 6(c) (i.e., as the popularity of the files decreases) the mean download time increases. When we consider a bundle of four files (experiment 5, extreme right in 6(c)) the mean download time is 405s. The mean download time of the bundle is larger than the mean download time of 329s experienced for file 1 in isolation but smaller than the mean download times for files 2, 3 and 4 in isolation. These results are explained as follows. File 1 is the most popular and stands little to gain in availability, so the cost of downloading more content outweighs the availability benefit of bundling. However, for the less popular files 2, 3 and 4, bundling reduces the download time by keeping the swarm self-sustaining during periods of publisher unavailability. In summary, if contents have different popularities, bundling

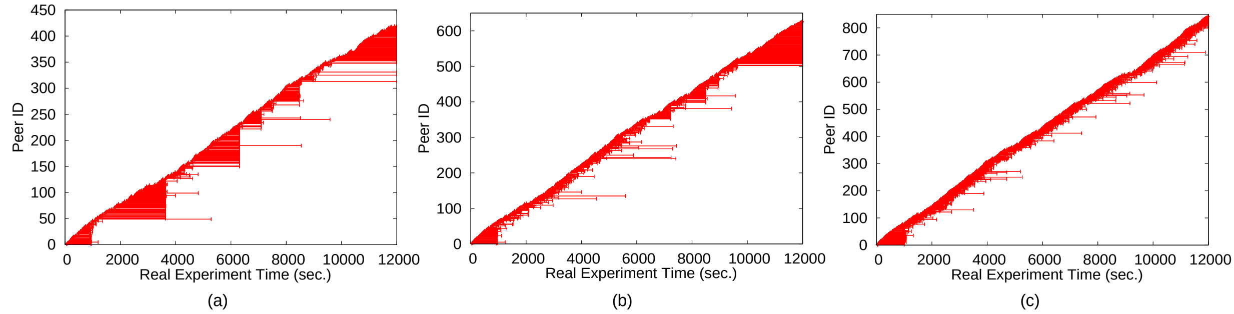

Figure 5: An intermittent publisher: (a) K=2; (b) K=3; (c) K=4. Each line starts when a peer arrives and ends when it leaves. As K increases, blocking probability decreases.

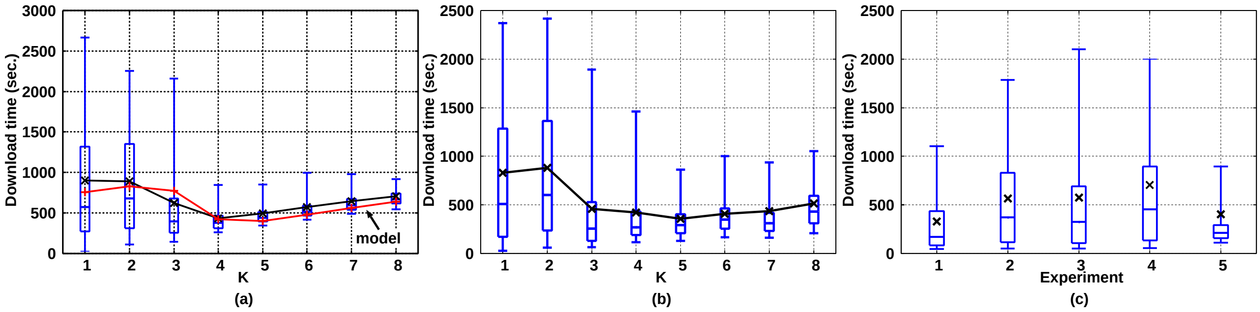

Figure 6: Download time versus bundling strategy. (a) exponential up and down times; (b) heterogeneous upload rates; (c) heterogeneous demand ($\lambda_i = \frac{1}{8i}$, $i = 1, \ldots, 4$), files bundled in experiment 5.

may increase the download times of peers downloading the most popular contents but can benefit those downloading unpopular files. In this example, bundling slightly increases the download times of 48% of peers who download the most popular content but significantly benefits the majority of the population.

4.3.4 Arrival patterns

Our model as well as experiments so far assumed Poisson peer arrivals at a steady rate. To evaluate the sensitivity of our conclusions to the Poisson assumption, we repeated experiments similar to those in Figure 6 using scaled versions of real arrival patterns observed in our measurement traces collected in §2. We found that using trace-driven arrivals does not qualitatively change our conclusions (refer to [10] for details).

However, we believe our model's conclusions may not hold if the mean arrival rate is not steady for a long enough duration of time. In particular, our model will overestimate the length of the busy period and consequently availability if the arrival rate decreases significantly before the end of the busy period determined by the current arrival rate. Nevertheless, we found a significant number of swarms with relatively steady arrival rates in our measurement traces. For example, out of the 1,155 swarms associated with the TV show "Lost", 911 were published more than one month before we started our measurement. Figure 7(a) shows a typical new swarm in its first month and a typical old swarm after two years of its creation. The arrival rates of old swarms show much less variation compared to the arrival rates of new swarms. Our model can be used to predict the availability, download times, and the impact of bundling for such swarms.

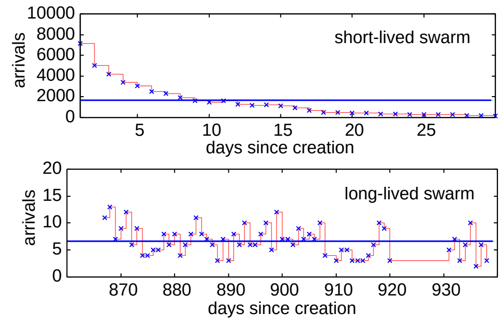

Figure 7: Typical peer arrival patterns of short-lived and long-lived swarms.

5. RELATED WORK

A large body of prior work has studied availability, performance and incentive issues in BitTorrent [3]. To our knowledge, this paper presents the first analytical model for content availability in BitTorrent-like swarming systems. We were also unable to find prior work studying the availability and performance implications of bundling in BitTorrent.

Ramachandran et al. [18] study the blocked leecher problem, where extant as well as arriving peers may have to wait for a long period of time for some blocks of the file that are no longer available. To address the problem, they propose BitStore, a token-based incentive architecture to obtain the missing blocks cached at other peers that had previously downloaded the file.

Neglia et al. [13] perform a large-scale measurement study to investigate availability in BitTorrent. They find that tracker availability is a serious enough problem that many torrents use replicated or DHT-based trackers for fault tolerance. Our focus is not on the availability of the tracker (or control plane), but on content availability (or data plane).

Both Susitaival et al. [20] and Wong et al. [22] relate the busy period of the M/G/∞ queue to content availability in BitTorrent. Menasche et al. [12] model the availability of chunks in a swarm using a set of M/G/∞ queues in tandem, under assumptions similar to those of coupon collector systems. Our model differs from [20, 22, 12] in two ways as it 1) quantifies content availability while accounting for publisher dynamics, and 2) quantifies the impact of bundling on availability and download time.

In the context of enterprise swarming, Menasche et al. [11] studied the strategic interaction between publishers, who are always available, and peers. Publishers control their pricing and bundling strategies while peers decide which content to download. The authors show that in the monopoly there always exist a unique equilibrium between the single publisher and the peers.

Qiu and Srikant [17] building upon earlier work by Veciana et al. [21] present a fluid model to analyze the download time performance of BitTorrent in steady-state. In contrast, our model accounts for both performance and availability similar in spirit to *performability* [7]. A naive adaptation of the fluid model in [17] to bundles suggests strictly longer download times under bundling, whereas our model shows that bundling can decrease download times by improving availability.

Many recent works have studied performance and fairness of a single swarm [8, 9, 1, 4]. Collaboration across swarms was studied by Guo et al. [6] suggesting many unexplored inter-torrent opportunities for block exchanges. Piatek et al. [15] suggest that propagating peer reputations limited to one hop can incent exchanges across swarms. Sirivianos et al. [19] propose an architecture where a commercial content provider provides "credits" to incent more cooperation between peers. Bundling is complementary to inter-swarm collaboration based on micropayment schemes to improve content availability. Micropayment schemes require a central bank to enable transactions and a tracking mechanism across swarms for peers to locate each other. In contrast, bundles are easy to set up and require no change to existing trackers or clients and is already in widespread use.

Economics of bundling.

Product bundling is a common commercial marketing strategy. The economics literature distinguishes between two forms of bundling [5]. In pure bundling or tying, a consumer can purchase the entire bundle or nothing at all. In mixed bundling, consumers have a choice to select parts of the package.

Both forms of bundling exist and have their pros and cons in BitTorrent's "bandwidth market" as well. Publishers can implement pure bundling by distributing bundled content as a zip archive. By forcing peers to download the whole bundle, pure bundling may make unpopular files more available, while subsidizing bandwidth costs for the publisher. However, it can delay those seeking exclusively popular files by forcing them to download content they do not want.

Mixed bundling is more common and can also improve availability. Publishers typically bundle files according to user interests, thus bundling can serve as a mutually beneficial recommendation system. A user seeking one episode of a TV show may decide to fetch the entire season for possible future viewing. A publisher might recommend a movie as part of a bundle to a user who may preview it and choose to pay for it after all [16, 22]. Even a small fraction of users opting to download more content than they strictly sought can significantly improve availability. Both mixed and pure bundling in BitTorrent have a beneficial side-effect: they replicate unpopular or rare content implicitly increasing their durability in the long run, i.e., it reduces the likelihood of rare content being lost permanently.

Our work opens up several avenues of future work. First, bundling may increase the traffic in the network, which motivates studying the implications of bundling for ISP pricing as well as its impact on content locality. Second, although our work sheds some light on what files make good candidates for bundling, more work is needed to understand how a content provider should optimally bundle files to meet performance or cost objectives, especially when the demand for a bundle may be different from the aggregate demand for its constituent files.

6. CONCLUSIONS

Peer-to-peer swarming in BitTorrent scales impressively to tolerate massive flash crowds, but falls short on availability. Although it is commonly observed that BitTorrent accounts for up to half of all Internet traffic today, it is less well known that half of the swarms are unavailable half of the time—an observation that does not bode well for the increasing commercial interest in integrating swarming with server-based content dissemination. Our work is a first step towards developing a foundational understanding of content availability in swarming systems.

By viewing BitTorrent as a queueing system, we were able to model content availability. The model suggests two important implications for bundling of content, a common practice among swarm publishers today. First, bundling improves content availability. Second, when the publisher is highly unavailable, bundling reduces the download time experienced by peers to obtain unpopular content. The latter implication is particularly intriguing as peers take less time to download more content. Although the model makes several simplifying assumptions, we were able to empirically validate its conclusions through large-scale controlled experiments with the mainline BitTorrent client over Planetlab.

7. ACKNOWLEDGEMENT

We thank Peter Key, John Lui, Giovanni Neglia and Shlomo Zilberstein for fruitful discussion on bundling. This work was supported in part by the NSF under award numbers CNS-0519922 and CNS-0721779. Research of D. Menasche also funded by a scholarship from CAPES/Fulbright (Brazil).

8. REFERENCES

[1] Bharambe, A. R., Herley, C., and Padmanabhan, V. N. Some observations on Bittorrent performance. In *SIGMETRICS* (2005).

[2] Browne, S., and Steele, J. Transient behavior of coverage processes with applications to the infinite server queue. *J. Appl. Prob. 30 (3)* (1993), 589–601.

[3] Cohen, B. Incentives build robustness in Bittorrent. In *P2PEcon* (2003).

[4] Fan, B., Chiu, D., and Lui, J. Stochastic analysis and file availability enhacement for Bittorrent-like sharing systems. In *IWQoS* (2006).

[5] Fuerderer, R., Herrmann, A., and Wuebker, G. *Optimal Bundling*. Springer, 1999.

[6] Guo, L., Chen, S., Xiao, Z., Tan, E., Ding, X., and Zhang, X. A performance study of Bittorrent-like peer-to-peer systems. *JSAC 25(1)* (2007), 155–169.

[7] Haverkort, B. R., Marie, R., and Rubino, G. *Performability Modelling*. Wiley, 2001.

[8] Legout, A., Liogkas, N., and Kohler, E. Rarest first and choke algorithms are enough. In *IMC* (2006).

[9] Massoulie, L., and Vojnovic, M. Coupon replication systems. In *SIGMETRICS* (2006).

[10] Menasche, D., Rocha, A., Li, B., Towsley, D., and Venkataramani, A. Content availability and bundling in swarming systems. *University of Massachusetts, Amherst, Technical Report UM-CS-2009-010* (2009).

[11] Menasche, D. S., Neglia, G., Towsley, D., and Zilberstein, S. Strategic reasoning about bundling in swarming systems. In *GameNets* (2009).

[12] Menasche, D. S., Rocha, A., de Souza e Silva, E., Leao, R. M. M., Towsley, D., and Venkataramani, A. Modeling chunk availability in peer-to-peer swarming systems. In *Performance Evaluation Review* (2009).

[13] Neglia, G., Reina, G., Zhang, H., Towsley, D., Venkataramani, A., and Danaher, J. Availability in BitTorrent systems. In *Infocom* (2007).

[14] Piatek, M., Isdal, T., Anderson, T., Krishnamurthy, A., and Venkataramani, A. Do incentives build robustness in Bittorrent? In *NSDI* (2007).

[15] Piatek, M., Isdal, T., Krishnamurthy, A., and Anderson, T. One hop reputations for peer to peer file sharing workloads. In *NSDI* (2008).

[16] Pucha, H., Roy, S., and Hu, Y. C. Take one get one free; leveraging peer-to-peer networks for content promotion. In *Global Internet Symposium* (2006).

[17] Qiu, D., and Srikant, R. Modeling and performance analysis of Bittorrent-like peer-to-peer networks. In *SIGCOMM* (2004).

[18] Ramachandran, A., das Sarma, A., and Feamster, N. Bitstore: An incentive compatible solution for blocked downloads in Bittorrent. In *NetEcon* (2007).

[19] Sirivianos, M., Park, J. H., Chen, R., and Yang, X. Free-riding in Bittorrent networks with the large view exploit. In *IPTPS* (2007).

[20] Susitaival, R., Aalto, S., and Virtamo, J. *Lecture Notes in Computer Science*. 2006, ch. Analyzing the dynamics and resource usage of peer-to-peer file sharing by a spatio-temporal model.

[21] Veciana, G., and Yang, X. Service capacity of peer-to-peer networks. In *Infocom* (2004).

[22] Wong, S., Altman, E., and Ibrahim, M. Peer to peer networks: interplay between legislation and information technology. In *INRIA Technical Report 6889* (2009).

APPENDIX

Background

Our results rely on those reported by Browne and Steele [2] on the busy period of an M/G/∞ queue where the customer initiating the busy period has an exceptional residence time.

Let customers arrive according to a Poisson process with rate β. If we allow customers initiating a busy period to draw their residence times from a distribution $H(\cdot)$ with Laplace transform $h(\cdot)$ and mean θ while all other customers draw their residence times from a distribution $G(\cdot)$, the expected busy period length is given by

$$E[B] = \theta + \sum_{i=1}^{\infty} \frac{\beta^i}{i!} \int_0^{\infty} (1 - H(x)) \Big[\int_x^{\infty} \Big(1 - G(u)\Big) du \Big]^i dx \quad (17)$$

When $G(x) = 1 - e^{-x/\alpha}$, i.e., all customers except the first draw their service times from an exponential distribution with mean α, the equation above reduces to

$$E[B] = \theta + \sum_{i=1}^{\infty} \frac{(\beta\alpha)^i \alpha [1 - h(i/\alpha)]}{i! i} \quad (18)$$

If the customer initiating a busy period also draws its service time from an exponential distribution,

$$E[B] = \theta + \alpha\theta \sum_{i=1}^{\infty} \frac{(\beta\alpha)^i}{i!(\alpha + i\theta)} \quad (19)$$

Finally, if $\theta = \alpha$,

$$E[B] = (e^{\beta\alpha} - 1)/\beta \quad (20)$$

Proofs

Throughout the proofs, let $\hat{\lambda} = \max\{\lambda_k\}$, $\check{\lambda} = \min\{\lambda_k\}$, $\hat{s} = \max\{s_k\}$, $\check{s} = \min\{s_k\}$.

Derivation of equation (9)

Proof. We use equation (17) to obtain (9). Let the download time of customers that arrive during the busy period be given by

$$X = \begin{cases} X_1 & \text{with probability } q_1 \\ X_2 & \text{with probability } q_2 = 1 - q_1 \end{cases}$$

where $E[X_i] = \alpha_i$. Then,

$$G(u) = 1 - q_1 e^{-\frac{1}{\alpha_1} u} - q_2 e^{-\frac{1}{\alpha_2} u} \quad (21)$$

and

$$E[B] = \theta + \sum_{i=1}^{\infty} \frac{\beta^i}{i!} \int_0^{\infty} I(z, i) dz \quad (22)$$

where

$$I(z, i) = (1 - H(z)) \sum_{j=0}^{i} \binom{i}{j} \Big[\Big(q_1 \frac{e^{-\frac{1}{\alpha_1} z}}{\frac{1}{\alpha_1}} \Big)^j \Big(q_2 \frac{e^{-\frac{1}{\alpha_2} z}}{\frac{1}{\alpha_2}} \Big)^{i-j} \Big] \quad (23)$$

Substituting (23) into (22) and using integration by parts yields

$$E[B] = \theta + \sum \frac{\beta^i}{i!} \sum_{j=0}^{i} \binom{i}{j} \frac{q_1^j}{\frac{1}{\alpha_1^j}} \frac{q_2^{i-j}}{\frac{1}{\alpha_2^{i-j}}} \left[\frac{1 - h(\frac{j}{\alpha_1} + \frac{i-j}{\alpha_2})}{\frac{j}{\alpha_1} + \frac{i-j}{\alpha_2}} \right] \quad (24)$$

and if customers initiating a busy period draw their service times from an exponential distribution with mean θ, $h(s) = \theta^{-1}/(\theta^{-1} + s)$, which yields (9). □

Proof of Lemma 3.1

PROOF. Since all the terms in B_k, $k = 1, \ldots, K$, are lower bounded and upper bounded by terms that do not depend on K, $E[B_k]$ is bounded hence $E[B_k] = \Theta(1)$.

To show that $\log E[\mathcal{B}] = \Theta(K^2)$ we consider a special process where the following conditions hold,

- during the busy period, customers arrive with rate β, where $\beta = R + \sum_{k=1}^{K} \lambda_k$,
- the residence times of all customers, including the first, arriving in a busy period are drawn from an exponentially distributed random variable with mean α,
- α and β are upper bounded by

$$\alpha \leq K\hat{s}/\mu \qquad \beta \leq K\hat{\lambda} + R \quad (25)$$

- α and β are lower bounded by

$$\alpha \geq K\check{s}/\mu \qquad \beta \geq K\check{\lambda} \quad (26)$$

The average busy period, $E[\mathcal{B}^\star]$, of the special process is given by (20)

$$E[\mathcal{B}^\star] = (e^{\beta\alpha} - 1)/\beta \quad (27)$$

First, we show that $\log E[\mathcal{B}^\star] = O(K^2)$.

$$E[\mathcal{B}^\star] \leq \Big[\exp(K^2 \hat{s}\hat{\lambda}/\mu + o(K^2)) - 1\Big] \Big/ (K\check{\lambda}) \quad (28)$$

$$\lim_{K\to\infty} \frac{\log E[\mathcal{B}^\star]}{K^2} < \infty \quad (29)$$

Next, we show that $\log E[\mathcal{B}^\star] = \Omega(K^2)$.

$$E[\mathcal{B}^\star] \geq \Big[\exp(K^2 \check{s}\check{\lambda}/\mu) - 1\Big] \Big/ (K\hat{\lambda} + R) \quad (30)$$

$$\lim_{K\to\infty} \frac{K^2}{\log E[\mathcal{B}^\star]} < \infty \quad (31)$$

Therefore, $\log E[\mathcal{B}^\star] = \Theta(K^2)$.

To extend the result above to the parameterization of $E[\mathcal{B}]$ made in Section 3.3.1 we proceed as follows. For the upper bound, $\log E[\mathcal{B}] = O(K^2)$, consider a modified process in which the residence times of all customers arriving during a busy period are drawn from an exponential random variable with mean $\alpha = \max(U, K\hat{s}/\mu)$. Denote the busy period of the modified process by $\widehat{B}$. For K sufficiently large, $\alpha = K\hat{s}/\mu$. Noting that conditions (25) hold it follows from (28)-(29) that $\log E[\widehat{B}] = O(K^2)$. Since $E[\widehat{B}] \geq E[\mathcal{B}]$, $\log E[\mathcal{B}] = O(K^2)$.

For the lower bound, $\log E[\mathcal{B}] = \Omega(K^2)$, consider a modified process in which the residence times of all customers arriving during a busy period are drawn from an exponential random variable with mean $\alpha = (U, K\check{s}/\mu)$. Denote the busy period of the modified process by $\widetilde{B}$. For K sufficiently large, $\alpha = K\check{s}/\mu$. Therefore, conditions (25)-(26) hold and (30)-(31) imply that $\log E[\widetilde{B}] = \Omega(K^2)$. Since $E[\widetilde{B}] \leq E[\mathcal{B}]$, $\log E[\mathcal{B}] = \Omega(K^2)$.

Finally, given that $E[\mathcal{N}] = E[\Lambda\mathcal{B}]$ we also have

$$\log E[\mathcal{N}] = \Theta(K^2) \quad (32)$$

From (28) and (30), the mean number of customers served in a busy period of the special process, $E[\mathcal{N}^\star]$, satisfies

$$K\check{\lambda}\frac{\exp(K^2\check{s}\check{\lambda}/\mu)-1}{K\hat{\lambda}+R} \leq E[\mathcal{N}^\star] \leq \hat{\lambda}\frac{\exp(K^2\hat{s}\hat{\lambda}/\mu + o(K^2))-1}{\check{\lambda}} \quad (33)$$

which yields (32) applying an argument similar to the one used to show that $\log E[\mathcal{B}] = \Theta(K^2)$. □

Proof of Theorem 3.1

PROOF. Since all the terms in P_k, $k = 1, \ldots, K$, are lower bounded and upper bounded by terms that do not depend on K, $E[B_k]$ is bounded hence $P_k = \Theta(1)$.

We rewrite $-\log \mathcal{P}$ as

$$\begin{aligned} -\log \mathcal{P} &= -\log \frac{1/R}{E[\mathcal{B}] + 1/R} \\ &= -\log(1/R) + \log(e^{\Theta(K^2)} + 1/R) \end{aligned}$$

where the last equality follows from Lemma 3.1.

We now show that $-\log \mathcal{P} = \Theta(K^2)$. First, we show that $-\log \mathcal{P} = O(K^2)$,

$$\lim_{K\to\infty} \frac{-\log \mathcal{P}}{K^2} = \kappa_1 + \lim_{K\to\infty} \frac{\log(e^{\Theta(K^2)} + 1/R)}{K^2} < \infty \quad (34)$$

Then, we show that $-\log \mathcal{P} = \Omega(K^2)$,

$$\lim_{K\to\infty} \frac{K^2}{-\log \mathcal{P}} = \lim_{K\to\infty} \left[\kappa_2 + \frac{\log(e^{\Theta(K^2)} + 1/R)}{K^2}\right]^{-1} < \infty \quad (35)$$

from which we conclude that $-\log \mathcal{P} = \Theta(K^2)$. □

Proof of Lemma 3.3

PROOF. Due to the memoryless property of the exponential random variable, the virtual customer that starts the residual busy period is characterized by a random variable $Y = \max\{X_1, \ldots, X_n\}$ where $X_1, \ldots, X_n$ are exponential random variables with mean s/μ. Therefore, Y is an hypoexponential distribution with parameters $(s/\mu, s/(2\mu), \ldots, s/(n\mu))$, which has Laplace transform $\prod_{i=1}^{n}(i\mu/s)/(s + i\mu/s)$ and mean $\sum_{i=1}^{n} s/(i\mu)$. Equation (18) can be used to compute $B(n, 0)$ for any value of n (eq. (12)).

Let us denote by $T_{i,j}$ the time it takes for a residual busy period which starts with i peers to reach a population size of $j < i$ peers, where $B(i, j) = E[T_{i,j}]$. For $n > l$ and $n > k > l$, we have that $T_{n,l} = T_{n,k} + T_{k,l}$. Therefore, in general $E[T_{n,l}] = E[T_{n,k}] + E[T_{k,l}]$ and in particular, $E[T_{n,l}] = E[T_{n,0}] - E[T_{l,0}]$. Equation (12) and $B(n, l) = B(n, 0) - B(l, 0)$ provide a way to compute $B(n, l)$ for arbitrary values of n and $l < n$. □

The reader can find the proofs of the other results presented in this paper in [10].