Characterization of BitTorrent swarms and their distribution in the Internet

Tobias Hoßfeld [a,*], Frank Lehrieder [a], David Hock [a], Simon Oechsner [a], Zoran Despotovic [b], Wolfgang Kellerer [b], Maximilian Michel [b]

[a] *University of Würzburg, Institute of Computer Science, Würzburg, Germany*
[b] *DOCOMO Communications Laboratories Europe GmbH, Munich, Germany*

A B S T R A C T

The optimization of overlay traffic resulting from applications such as BitTorrent is a challenge addressed by several recent research initiatives. However, the assessment of such optimization techniques and their performance in the real Internet remains difficult. Despite a considerable set of works measuring real-life BitTorrent swarms, several characteristics of those swarms relevant for the optimization of overlay traffic have not yet been investigated. In this work, we address this lack of realistic swarm statistics by presenting our measurement results. In particular, we provide a statistical characterization of the swarm sizes, the distribution of peers over autonomous systems (AS's), the fraction of peers in the largest AS, and the size of the shared files. To this end, we consider different types of shared content and identify particular characteristics of regional swarms. The selection of the presented data is inspired by ongoing discussions in the IETF working group on application layer traffic optimization (ALTO). Our study is intended to provide input for the design and the assessment of ALTO solutions for BitTorrent, but the applicability of the results is not limited to that purpose.

© 2010 Elsevier B.V. All rights reserved.

1. Introduction

Overlay traffic resulting from applications such as BitTorrent emerges as a high burden for network operators today. The problem arises of how to effectively control and manage such traffic stemming from end-to-end overlay applications from within the network. Recently, this challenge is addressed by research initiatives like SmoothIT [1] and P4P [2] and possible solutions such as an Oracle [3] are proposed. Furthermore, its importance has triggered standardization and the ALTO working group of the IETF has been founded in November 2008.

Locality awareness is considered as a straightforward solution by all current research initiatives. Traffic generated by overlay applications typically crosses borders of network operator domains (so called autonomous systems, AS's) multiple times and uses a larger number of links than

required because the overlays are unaware of the underlying physical network properties. In this way, it causes high costs for the network providers and leads to an inefficient usage of the physical network resources. At the same time, BitTorrent might offer suboptimal performance as seen from its users' point of view since long distance connections are also more likely to offer less transfer capacity. The concept of locality awareness is to optimize the traffic flow with information about the location of a content providing peer in the underlying network. For example, any peer might be provided with a list of peers for download that are marked according to the position inside or outside the AS of the requesting peer. Thus, quality of service and network usage can be optimized at the same time, for the benefit of the overlay application and the network provider.

Several implementation options of locality awareness for BitTorrent-based P2P networks have been proposed and their performance was evaluated mostly in simulation studies, e.g., [2,4,5], or using experiments in controlled

---

* Corresponding author.
  E-mail address: hossfeld@informatik.uni-wuerzburg.de (T. Hoßfeld).

environments, e.g., [2,6]. First evaluations using a uniform distribution of peers over AS's revealed that locality awareness is able to reduce inter-domain traffic without deteriorating the performance for the P2P users in most scenarios [4,5]. However, recent studies [6–9] have shown that this is different under more realistic conditions, in particular when some AS's contain more peers than others. For example, peers in AS's with a high number of peers experience a performance degradation when preferentially communicating with peers in the same AS. This can be avoided by a refinement called "partition merging strategy" [6] of the original mechanisms. In [8] we investigated two implementation options of locality awareness in scenarios based on the swarm characterizations presented here. While one implementation improves the performance for peers in AS's with a high number of peers, the other implementation decreases the performance for the same peers. Therefore, we conclude that realistic scenarios are required to accurately assess the performance of locality awareness. Appropriate measurement results for this purpose however are not available to the research community up to now and we extend existing studies by providing such data in this paper. Note that we do not investigate in which way our results affect current proposals for locality awareness. The reason is that the performance depends heavily on the concrete implementation of the mechanisms, i.e., the performance of similar or slightly modified mechanisms can differ considerably in the same scenario [6,8]. Instead, we present possible effects motivating the relevance of the parameters we measure.

In this study we report results of our large-scale measurement study of live BitTorrent swarms and derive important characteristics relevant for traffic optimization in overlay networks. The measurement results comprise a comprehensive set of swarms for different types of content listed at the index servers mininova.org and pirate-bay.org. We have measured the swarm size, swarm dynamics in terms of number of leechers and seeders, and the distribution of peers over AS's per swarm. We have also analyzed the details of individual swarms to understand content clustering (e.g., availability of certain content in specific regions only). The measurements have been performed from June 2008 to May 2009 using the PlanetLab [10] and G-Lab experimental facilities [11]. Some additional measurement results are provided in our technical report [12]. Based on these measurements and an additional public data set [13], we derive characterizations of the swarm size, the distribution of the peers over AS's, the fraction of peers in the largest AS, and the size of the shared files. In addition, we present multivariate correlation matrices of these parameters to show to which degree these values depend on each other. In particular, our characterization of BitTorrent swarms reflects that peers are not homogeneously distributed among AS's, but most of the peers are located in a small number of top AS's. Furthermore, we provide quantitative results on the skewness of the peer distribution based on the measurements.

The measurement results and the characterizations can serve as input for the performance evaluation of locality awareness in order to gain insights into the behavior of proposed solutions for traffic optimization under real-world conditions. In addition, they show the composition of a large set of swarms observed in the Internet, which can be used to assess the overall gain of a proposed solution if the gain achieved in some typical scenarios is known. In particular, they show that 80% of the BitTorrent peers are located in 20% of the swarms. Finally, a deeper understanding about AS-level properties of real BitTorrent swarms helps in refining current proposals and in designing new mechanisms.

The remainder of this paper is organized as follows. Related work is discussed in Section 2. We explain the measurement setup in Section 3 and provide the measurement results in Section 4. Based on these results, we present the corresponding statistical characterizations for BitTorrent swarms in Section 5. Finally, Section 6 summarizes this work.

2. Related work

Measuring and modeling of BitTorrent swarms have received considerable attention during the last years. In the following, we give an overview of the most prominent works on that topic and explain in which way our work differs from them. In addition, we present current proposals for traffic optimization in P2P networks and corresponding performance evaluations because they served as motivation for our choice of properties of BitTorrent swarms to measure.

2.1. Measurements and models of BitTorrent

In [14], the authors follow the lifetime of one specific torrent and analyze BitTorrent's main performance indicators (e.g., download times). Besides examining its download performance, [15] makes a step further toward providing measurements useful for the modeling of BitTorrent. The peer uptime distribution, their bandwidth distribution, peer arrival process properties as well as the distribution of seeders across time are the main quantities [15] focuses on. In [16], the authors provide models for several key parameters of BitTorrent networks such as the arrival process, seeding times, and downloading failures and build a graph-based multi-torrent model. The set of properties in the focus of our paper is not overlapping with these, i.e., the information we provide is complementary to the information provided by [14–16]. In contrast to these studies, we pay considerable attention in this paper to the distribution of peers in BitTorrent swarms across AS's.

2.2. Locality awareness solutions for BitTorrent

Locality promotion has been so far suggested in the literature as the main solution class for reducing inter-domain traffic. It requires peers to preferentially select neighbors from the same AS rather than those outside the AS when forming the overlay graph. Bindal et al. [4] and Aggarwal et al. [3] were the first to analyze how locality promotion can help to reduce the generated traffic and

improve the performance of BitTorrent and Gnutella, respectively. They both discover significant improvements of the application performance (i.e., reduction of download times) and reduction of cross-ISP traffic. A complementary implementation option to [4], which is especially useful if the number of peers per AS is low, is presented in [5]. Both, [4] and [5] use simulations with a uniform distribution of peers over AS's for the performance evaluation. P4P [2] takes a somewhat different approach. Network operators provide so-called iTrackers which communicate with the application trackers, e.g. a BitTorrent tracker, exchanging p-distance values. These values represent the application costs of the path between two peers and can be configured by the ISP. Consequently, the p-distance can also reflect intra-AS topologies or priorities for different inter-AS links. Choffnes and Bustamante [17] propose a method for localizing BitTorrent traffic without the need for an additional infrastructure such as iTrackers or an Oracle and evaluates this approach using a plugin called Ono for the open-source BitTorrent client Vuze. Ono is based on the idea that peers with a similar redirection behavior of content distribution network (CDN) servers are close to each other.

2.3. Measurement-based performance evaluations of locality awareness

Some recent works on the performance of locality awareness solutions for BitTorrent consider also scenarios with Internet-like distributions of peers over AS's. [6,7,18] crawl popular BitTorrent sites and download the metadata in form of .torrent files. Using a set of requests to the trackers they obtain the IP addresses of the peers participating in the swarms and associate them with an AS number. The measurements are used as input for the evaluations, but only a very small subset of the measurement results is presented in the papers. In [7] no measurement data is presented and in [6] the authors specify concrete values of the number of peers and AS's per swarm only for three *reference torrents*. [18] also uses the measurements mainly for the evaluation, but it provides cumulative distribution functions for the number of peers per swarm, the number of peers per ISP, and the number of swarms per ISP. Unlike these works we focus on the presentation of our measurement results. Therefore, we study a larger set of parameters and differentiate between different types of content. Furthermore, we provide statistical characterizations of the measured parameters which can serve as input for other performance evaluations.

In [6,8], uniform and Internet-like distributions of peers over AS's are used. Both studies show that the distribution of peers has an important impact on the performance of the locality awareness implementations for BitTorrent. In particular, with Internet-like peer distributions some of the peers can experience performance degradations in terms of reduced download speeds. To mitigate these negative effects, the original implementations need to be refined. Blond et al. [6] use a "partition merging strategy" and Lehrieder et al. [9] propose to group AS's with a small number of peers for that purpose. From this we derive two conclusions. First, small modifications or refinements of the implementations may change the behavior of the system considerably. Second, realistic scenarios are required in order to provide meaningful performance evaluation results. For that purpose, we present the results of our measurement study in this paper which can help to design new implementations or evaluation scenarios for locality awareness.

The work with the closest relation to ours is [19]. Wang et al. studied around 70,000 BitTorrent swarms from the btmon.org-BitTorrent site for 6 months in 2008, using 200 PlanetLab nodes with a customized BitTorrent client to retrieve the swarms' peer IP addresses. These IP addresses were run against the whois-service to resolve the IPs' autonomous systems. The paper mainly concentrates on swarms distributing video files, stating that video files show the highest regional (AS) interest, e.g., Chinese movies are mostly watched in China. The authors analyze the distribution of peers to AS's and conclude that in small swarms the application of locality awareness mechanisms is not useful, because the top AS of the swarm holds a large fraction of the whole swarm and the traffic is already naturally localized. On the other hand in large swarms the authors found no AS holding more than 6% of the whole swarm population, which makes the application of locality enhancements more favourable. Furthermore, they find that for large swarms the relation between ordered AS's of a swarm and the AS-fraction of a swarm (i.e., x-largest AS of a swarm – #peers in AS/#peers in swarm) follows the Mandelbrot–Zipf distribution. Eventually, the paper argues that AS's have a stationary property of forming a larger cluster within a swarm, and give a probabilistic approach how to predict the peers' membership in a large cluster. Peers in large clusters should apply locality aware neighbor selection, peers not in a large cluster should stay with the standard random neighbor selection. In contrast to this paper, we consider more media types in our measurement and not only the AS affiliation of the peers but also the country where the peers are located. We also cover more swarms from different torrent index servers. This allows us to generalize the results and to identify subgroups with special characteristics. Thus, we also provide a more diverse view on regional content, which is mentioned in [19] but not considered in detail. Especially, we show that the share of peers in one AS can be larger for regional content (as also mentioned in [6]) and provide a quantitative evaluation of how the distribution of peers over AS's of those swarms differs from the rest.

3. Measurement setup

The measurement setup described in this section aims at gathering data about live BitTorrent swarms from which we want to derive characterizations of the parameters relevant to locality awareness. First, we outline the BitTorrent protocol itself before introducing our measurement methodology.

3.1. The BitTorrent protocol

BitTorrent's objective is to disseminate one large file to a large number of users in an efficient way. For each file an

overlay network called *swarm* is created. According to the original BitTorrent specification, each overlay network consists of two different kinds of peers, the seeders and the leechers, and a so-called tracker. A *seeder* is a peer in the swarm that holds the complete file and uploads to others altruistically, whereas a *leecher* is still downloading the file.

For each swarm, a centralized component, the so-called *BitTorrent tracker*, stores information about the file itself and all peers in the swarm. This information includes the file size, the number of seeders and leechers, as well as the IP addresses of the peers. A peer joining the network asks the tracker for a list of active peers in the overlay. The tracker then returns (a) the number of seeders S and leechers L and (b) a random subset of k peers, i.e., k different IPs, to the requesting peer. Most trackers return $k = 50$ peers per default.

In order to avoid congestion at the tracker, the request rate of an individual peer is limited. The default value in the original BitTorrent tracker implementation from Cohen allows a single request every 5 min. However, in the Internet, various tracker implementations exist and in our measurements we have been able to contact various trackers every 10 s if necessary.

For searching files to download through the BitTorrent protocol, there are several websites that list indexes and directories of `.torrent` files. Such a website is referred to as a *torrent index*. A torrent index maintains a list of `.torrent` files containing metadata about the files to be shared and about the tracker, as well as additional information about the popularity of a file (in terms of number of seeders and leechers) or the date when the file was published.

3.2. Conducted measurements

To gain a more diverse view on the characteristics of existing swarm types than in the known work, we chose specific sets of swarms to measure. These are defined by a number of selection criteria which serve to define a number of swarm classes. In contrast to [19], we do not only analyze swarms found on one index and only distributing videos. Instead, we expand the insights gained from observing these swarms to other classes of swarms as well. According to a certain selection criterion and the desired type of content, the `.torrent` files are downloaded from a torrent index. As *selection criteria*, we consider (a) all available torrents, (b) the most popular torrents in terms of number of peers in the swarm, and (c) the most recent files which have been published in the last 24 h. As *type of content*, we distinguish between (1) music files, (2) TV series, (3) movies, (4) so-called "regional" movies which are in a certain language (German, Spanish, French, Italian, Dutch), and (5) all media independent of the type of content. These types are based on the user classifications at the torrent index servers. The considered *torrent index servers* cover the currently most popular ones in the Internet, (i) PirateBay, (ii) Mininova, and (iii) Demonoid. Here, the criteria (a)(3) and (a)(4) correspond to the class of swarms evaluated in [19]. Thus, we additionally consider other content types and indexes as well as specific subsets of swarms.

Table 1 summarizes the measurement experiments conducted over the period from June 2008 to May 2009. Each measurement experiment is assigned a unique identifier ID. which is used when describing the measurement results. In particular, we measure in each experiment the swarm size, the swarm dynamics, and the distribution of peers over AS's ('peer-dist.'). In order to measure the total number N of peers in a swarm and their corresponding AS's, we contacted the tracker and requested a list of peers. As a result, the number of seeders S and leechers L, and a set of k different IP addresses of peers are returned.

Since a tracker typically returns $k = 50$ IP addresses for a single request, we used a large number of machines with BitTorrent clients running on each of them. They contact the tracker simultaneously in order to get the IP addresses from all peers in the swarm at a single time instant, i.e., a snapshot of the swarm. In particular, several requests are sent within 5 min from all 219 nodes in PlanetLab [10] and 153 nodes in G-Lab [11], respectively, until $N = S + L$ different IP addresses are obtained. Then, the IP addresses are mapped to the origin AS using the RIPE database [20]. This measurement method is referred to as *distributed monitoring* in the remainder of the paper. However, for measuring the swarm size only, it is sufficient to monitor the tracker (denoted as 'tracker monitored' in Table 1 for setups Pop. and 24 h.) or to parse the website of the torrent index ('website parsed'), as done in experiment TV. Additionally, we consider a publicly available data set from Khirman [13] with measurement results of the swarm sizes of torrents on different torrent index servers (KPi., KDe., and KMi.). With all three methodologies ('website parsed', 'tracker monitored', and 'distributed monitoring') we can measure the swarm size. The distribution of peers over AS's can only be measured in those experiments where we used distributed monitoring of the tracker (cf. columns 'methodology' and 'observed' in Table 1) which is an extension of the method 'tracker monitored'.

To study the time dynamics of a swarm, several samples of the swarm size and the distribution of peers over AS's are captured over a longer period of time which is denoted as "*xx* samples every *yy* hours" instead of "snapshot" in the column "measurement per swarm" in Table 1. In that case, for example the average swarm size over this period of time is given, which may result in a decimal number, while a snapshot of a swarm always returns an integer value.

The different data sets describe the BitTorrent swarms under considerations with a different level of detail (Table 1). For example, the distribution of peers over AS's is only studied for the experiments performed in April 2009, i.e., Grp., Mov., Mus., Reg. and Ele. The reason is that we started with a rather basic methodology ('website parsed') in June 2008 and improved it during the course of this work. Therefore, we are not able to present the distribution of peers over AS's for the experiments TV., Pop., and 24 h., and this information is also not contained in the data sets KPi., KDe., and KMi. which we took from [13]. Furthermore, data about the change in the number of peers over time is only available for the experiments TV., Grp., and Ele. This is partially owed to feasibility reasons, in particular for the Mov. and Mus. experiments the number of swarms was to high to take hundreds of samples of the

Table 1
Overview on conducted measurement setups.

ID	Torrent index	Selection criteria	Type of content	Meas. per swarm	#Torrents	Methodology	Observed	Meas. date
TV.	PirateBay	All available	TV series	96 samples over 36 h	63,867	Website parsed	Swarm size	June 2008
Pop.	PirateBay	Most popular	Movies	Snapshot	4463	Tracker monitored	Swarm size	March 2009
24 h.	PirateBay	Last 24 h	All media	Snapshot	1048	Tracker monitored	Swarm size	March 2009
Grp.	Mininova	Groups w.r.t. size & language	Movies	440 samples over 88 h	16	Distributed monitoring	Swarm size and peer-dist.	April 2009
Mov.	Mininova	All available	Movies	Snapshot	126,050	Distributed monitoring	Swarm size and peer-dist.	April 2009
Mus.	Mininova	All available	Music	Snapshot	135,679	Distributed monitoring	Swarm size and peer-dist.	April 2009
Reg.	PirateBay	Top 30	Regional movies	Snapshot	120	Distributed monitoring	Swarm size and peer-dist.	May 2009
KPi.	PirateBay	All available	All media	Snapshot	1,682,355	Data taken from [13]	Swarm size	March 2009
KDe.	Demonoid	Community selected titles	All media	Snapshot	11,759	Data taken from [13]	Swarm size	March 2009
KMi.	Mininova	Legal torrents promotion	All media	Snapshot	4514	Data taken from [13]	Swarm size	March 2009
Ele.	Open movie "Elephants Dream"			8640 samples over 24 h	1	Distributed monitoring	Swarm size and peer-dist.	April 2009

swarm via distributed monitoring. While one needs to be aware of the aforementioned issues when interpreting the data, we suppose that their impact on the presented results is small and that the measurements remain comparable. For example, we will show for the Mov. and Mus. data sets that the IP addresses obtained via distributed monitoring are in good accordance with the number of peers obtained by tracker monitoring (cf. Fig. 1).

Some BitTorrent swarms exist without a tracker and are therefore called *tracker-less*. In these swarms the peers exchange the addresses of other peers in the swarm among each other using the peer exchange (PEX) protocol [21]. Since it is not possible to monitor those torrents with the aforementioned methodology, tracker-less torrents are not considered in this study.

3.3. Distributed monitoring of a tracker

The distributed monitoring of a BitTorrent tracker for obtaining the distribution of peers over AS's relies on experimental facilities, like PlanetLab [10] or G-Lab [11], with a large number of nodes. They are controlled by a central unit C which is located at the University of Wuerzburg in our measurements. C has established connections to the used PlanetLab and G-Lab nodes Ω. C is responsible for the distribution of the .torrent files to these monitoring nodes Ω, the initialization of the monitoring on Ω and the collection of the created result files from Ω. The monitoring on each node itself is realized with a python script that queries a tracker n times every t seconds. In our measurements, t is set to 15 s to avoid overloading the tracker, while n is chosen according to N, using the analysis described below.

In the following, we derive the number Y of required monitoring nodes in order to obtain all IP addresses of N peers in a swarm. Upon each request, the tracker returns a subset of $k = 50$ peers which are randomly chosen from all N peers. Denote by X the number of times the tracker has to be contacted to get N different IP addresses. The derivation of X is known as the *coupon collector's problem* [22]. In [23], we derived an exact solution which is given in the following.

Let $P(j, i)$ denote the probability to observe j different IPs after the ith tracker response. It is $P(j, i) = 1$ for $j \leqslant k$ and $i > 0$ since the first tracker response returns k different IPs. It is $P(j, i) = 0$ for $j > \min(ik, N)$, since a maximum of ik different IPs are retrieved after the ith tracker response and there are only N different IPs. This allows to recursively compute $P(j, i)$ for all other cases according to

$$P(j, i) = \sum_{m=0}^{k} \frac{\binom{j-m}{k-m} \cdot \binom{N-j+m}{m}}{\binom{N}{k}} \cdot P(j-m, i-1),$$
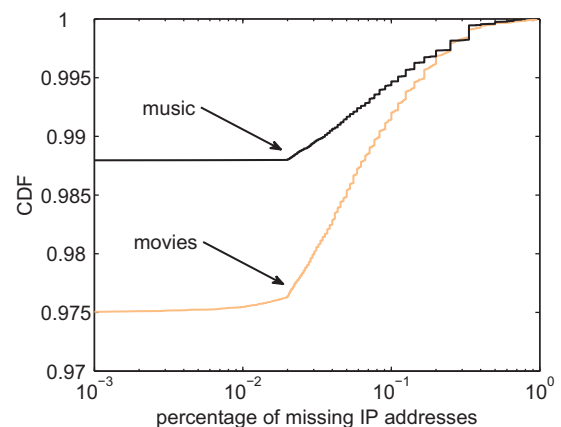
(1)

Fig. 1. CDF of the percentage of missing IP addresses.

which simply considers the number of possibilities to obtain $k - m$ old and m new IPs, normalized by the number of possibilities for k different IPs of a tracker response. As a result, we obtain the distribution X of the number of required tracker responses to get all N IPs which is $P(X = i) = P(N, i)$.

An upper bound of the average number of required tracker responses $E[X] = \sum_{i=0}^{\infty} iP(N, i)$ can be approximated [22] using the harmonic number $h_N = \int_0^1 \frac{1-x^N}{1-x} dx$,

$$E[X] \approx \frac{N \cdot h_N}{k}, \tag{2}$$

which is exact for $k = 1$. For example, to get a snapshot of the distribution of peers over AS's of a swarm with $N = 20{,}000$ peers, around $n = 20$ *requests* have to be sent from each of the 219 used PlanetLab nodes. This takes $n \cdot t = 5 minutes$. The computation of the number of tracker requests allows to estimate the required number of monitoring nodes and to adjust appropriately the parameters t and n, if a time frame of 5 min is allowed for capturing the snapshot.

However, it has to be noted that Eq. (2) only returns the average number of required tracker responses. Checking the percentage of missing IP addresses in our measurements, we observed that only for a small number of swarms some IP addresses are missing. In particular, we checked the percentage of missing IP addresses when observing the distribution of peers over AS's of a swarm Fig. 1 shows the cumulative distribution function (CDF) of the percentage of missing IP addresses when measuring the distribution of peers over AS's for the movies (Mov.) and music files (Mus.). For 97.5% of all movies (Mov.) and more than 98.5% of all music files (Mus.), all IP addresses in the swarm were captured. For the Reg. data set, which contains 120 swarms, all IP addresses are available for 118 swarms and in the Grp. data set we have them for all swarms. A reason for missing IPs is the fact that peers may go offline during the measurement interval of 5 min. This has no effect on the numerical values or on the conclusions.

To conclude this section, we describe as a side note one peculiarity we discovered during our measurement study. In our measurements, we found one swarm (Ele.) for which we discovered only 10% of the peers. In particular, the tracker returned a swarm size of 400,000 peers, however, we only observed 30,000 IP addresses. We used 219 PlanetLab nodes and requested the tracker every 10 s from each machine over 24 h. Thus, we received more than one million tracker responses with 50 IPs. In that case, we should observe at least around 375,000 different IPs.

There are two possible reasons for this observation. (1) The tracker always returns the same IP addresses. This could be the case when locality awareness mechanisms are implemented by the tracker. However, this is not the case here; the nodes in PlanetLab are distributed worldwide. Thus, it seems reasonable that the random generator or the function which returns a random subset of all peers is wrongly implemented. (2) The tracker returns wrong information about the number of seeders and leechers in the swarm. Since this tracker hosts only a single file (Ele.), we cannot check this hypothesis using other

swarms hosted at the same tracker. Still, the second explanation seems more likely to be the case, but we cannot prove it without investigating the source code of this tracker. In both cases, the question arises how an ALTO mechanism can reliably monitor swarms for badly implemented trackers.

4. Measurement results

In this section, we describe the results from the measurements. We focus on observations where previous studies provide only a general impression or where the results for specific swarm types contradict the accepted knowledge. In particular, we are interested in the characteristics of the swarm size and its development over time. Additionally, we consider the distribution of peers over AS's and over different countries, the clustering of peers in AS's and the correlation between the number of peers in an AS and its AS degree since these parameters are assumed to have important implications for the viability of locality promoting mechanisms. Finally, we report our findings on content that is popular only in specific regions of the world and summarize our main findings as well as the limitations of this study.

4.1. Population sizes in swarms

First we take a look at the size of the measured swarms. For this purpose, we analyzed the seeder and leecher population of swarms for different content types, e.g., movies, TV shows and music files, which are registered at different BitTorrent index websites.

Fig. 2 shows the observed swarm sizes for the data sets TV., Pop., 24 h., Mov., Mus., KPi., KDe., and KMi. The distribution of the number of peers is similar for all data sets except for the 24 h. and Pop. set. An explanation for this divergence is the fact that these two sets feature swarms with specific characteristics due to the popularity of the shared content. While the Pop. set of swarms contains swarms with highly sought content by definition, it is a reasonable assumption that the recently added files of the 24 h. set are also more popular than the average since users are interested in new content which is available for the first time.
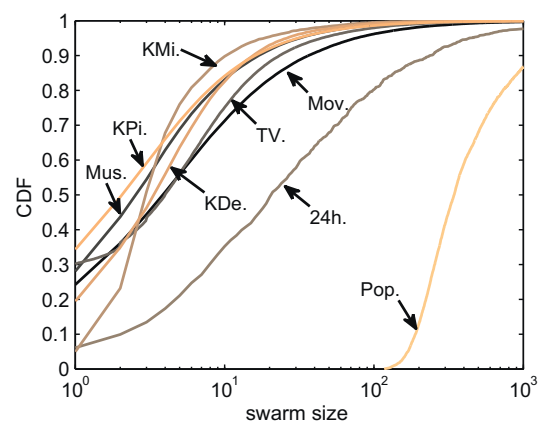

Fig. 2. CDF of the number of total peers in a BitTorrent swarm.

The according data for all measured data sets is given in Table 2. It contains the statistics for the total number of observed swarms, the mean value μ and coefficient of variation c_{var} of their sizes in terms of number of peers, the skewness, kurtosis and maximum of the swarm size distribution as well as the 95th percentile q_{95} both as an absolute value and normalized by the mean swarm size. Finally, the fraction of swarms π_{80} that contain 80% of the peers and the correlation $C(S,L)$ between the number of seeders and leechers in all swarms of the whole data set is shown.

The first observation we make about these results is that the swarm size depends on the content shared. This is in line with the observations for video file swarms from [19]. The swarms which distribute movies are the largest on average whereas smaller music files are shared by less peers on average. This can be attributed to the fact that larger files take longer to download, leading to a longer online time of peers and therefore a higher population in the swarm. This should be offset by the resulting additional upload bandwidth offered to the swarm. However, it can be shown analytically, e.g., by adapting the analysis of [24], that download times do increase in such swarms. A further reason for the larger swarms size could be that movie content is more popular than music.

The skewness and the kurtosis of the swarm sizes provide further insights into the distribution of the number of peers in the different data sets. They characterize to which degree some very large swarms are contained in the data sets. The column q_{95} in Table 2 contains the 95th percentile, which also characterizes the distribution of the swarm sizes. In particular, it shows the swarm size which is reached or exceeded by 5% of the swarms in the data set.

Regarding the different data sets, the coefficient of variation of the swarm size is in the same range, with the exception of the Khirman set of PirateBay swarms (KPi.). This set also differs significantly in terms of skewness, kurtosis and maximum swarm size. Although we cannot judge the source of this discrepancy with our data and the other data sets from Khirman, we still observe that at least the 95th percentile normalized by the mean value is comparable to the corresponding values for the other data sets.

Another general observation is that the Pareto principle holds for most of the evaluated data sets. The π_{80} value, i.e., the fraction of top swarms that contain 80% of all peers in all swarms of the set, is around 0.2 for all sets except the top movies and the Khirman data for the Mininova and Demonoid sites. This means that 80% of the peers belong to 20% of the swarms. It is clear that the most popular content as covered by the Pop. data set do not show this Pareto property since the different files here are equally popular and represent only a very specific part of the total shared content.

Finally, there is a strong correlation $C(S,L)$ between the number of seeders and the number of leechers in a swarm. This is intuitively clear since more leechers mean a larger number of potential seeders, and swarms with only a few seeders are normally not popular due to long download times.

From these observations we draw some conclusions how they could impact a locality aware mechanism. The type of shared content has an impact on the swarm size and therefore potentially on the effectiveness of different locality promoting solutions. We will see in the next sections that this is also true for the topological characteristics of a swarm, which also depend on the content shared. In general, the swarm size distribution is heterogeneous with a Pareto-like distribution of the total peer population on the different swarms. Also, recently released and popular content leads to much larger swarms in comparison.

In addition, there is a significant amount of very small swarms containing less than 40 peers. With typical BitTorrent client parameters, each peer in such a swarm will know all other peers because it tries to have at least 40 neighbors. The result is a fully meshed swarm. Consequently, accepted solutions using Biased Neighbor Selection (BNS) as introduced in [4], where peers close in the topology are preferred as neighbors, will probably have a low impact on these swarms since there is no choice to be made in the neighbor selection. On the other hand, the share of traffic that can be influenced by targeting only the comparably few top swarms, including new and popular content, is significant (around 80%, the corresponding estimation is presented in Section 4.4). The effort to do so is possibly much lower than when trying to cover all or at least most of the swarms because algorithms do not need to cope with special characteristics of small swarms. To optimize the monitoring of swarms in order to find these candidate swarms, it may help to just keep track of the seeder population since it is strongly correlated to the number of leechers and thus the total population of a swarm. These statements are not meant to be true in general and for every mechanism, they rather show examples how the data provided in this section can be important for the assessment of locality aware mechanisms.

Table 2
Statistics on the number of peers in a swarm.

ID	Swarms	μ	c_{var}	Skew.	Kurtosis	Max.	q_{95}	$\frac{q_{95}}{\mu}$	π_{80}	$C(S,L)$
Mov.	126,049	25.46	8.47916	51.8868	3573.01	20,079	76	2.98476	0.128672	0.8441
TV.	63,867	15.53	6.46545	29.45	1246.99	7276	45	2.8814	0.172076	0.7067
Mus.	135,679	9.76	4.24327	28.4287	1432.57	3813	32	3.2801	0.248071	0.6113
KPi.	1,682,355	11.12	13.4178	216.519	69248.6	72,988	31	2.78752	0. 177795	0.8526
KMi.	4514	6.99	3.1686	19.7756	535.817	763	19	2.71652	0.452813	0.5298
KDe.	11,759	9.73	4.64484	22.8985	663.787	1883	27	2.77568	0.3087	0.6506
Pop.	4463	691.14	2.07849	9.86953	144.064	30,691	2068	2.99218	0. 449698	0.7297
24 h.	1048	146.68	5.36564	17.2031	386.368	19,748	435	2.96556	0.118321	0.6535

4.2. Time-dynamics within a swarm

In this section, we investigate in which way the population of a swarm varies over time. The evolution of BitTorrent swarm populations during the whole life time of a swarm has already been analyzed in literature, e.g., in [14,16]. However, we focus here on a shorter time scale and investigate how fast the population typically grows or diminishes during our measurement period lasting 36 hours, i.e., one day and a half. In addition, we analyze which fraction of swarms is subject to diurnal fluctuations and how pronounced these fluctuations are.

For this section, we focus on the data set `TV.` since this contains more than 60,000 swarms and their temporal evolution. In order to illustrate some examples, we also consider the `Grp.` data set. However, this set contains only 16 swarms and is therefore less suitable for statistical analysis. For all other data sets we do not have measurements about the temporal evolution of the swarm sizes.

4.2.1. Increasing, constant, and decreasing swarms

While it may be efficient to promote locality in a swarm that was measured as being large at a given time instant, it may be less efficient when the swarm shrinks quickly after that snapshot. To gain insights into the time-dependent behavior of swarms, we measured 96 samples of the swarm sizes $n_i(s)$, $i \in \{1,96\}$ for every swarm s of the data set `TV.` The samples were equally distributed over 36 h. For all swarms, we calculate the average swarm size $\mu(s)$, the standard deviation $\sigma(s)$, and the coefficient of variation $c_{var}(s)$ of the 96 samples $n_i(s)$. In addition, we define the span of a swarm during the measurement period as $\Delta(s) = \max_i(n_i(s)) - \min_i(n_i(s))$. This metric represents the largest variation of the swarm population we observed in terms of peers. We call all swarms s with $\Delta(s) = 0$ constant swarms. The remaining swarms are increasing if their minimum value n_i has a lower index i than their maximum value. Otherwise, we denote them as decreasing.

We make the following observations in the data set `TV.`: All three groups (constant, increasing, and decreasing) contain almost the same fraction of swarms (33.81%, 32.88%, and 33.31%, respectively). However, the constant swarms are all very small (cf. Fig. 3). In addition, there is no significant difference between the CDFs of the sizes of increasing and decreasing swarms which is reasonable since the fact that a swarm is growing or shrinking is not correlated with its current size.

Fig. 4 shows CDFs for the span $\Delta(s)$ of the swarms normalized by the average swarm size over the 36 h time period. We observe that for only 10% of the swarms their span is below 60% of their average size and for 50% of the swarms it is higher than the average swarm size. Furthermore, the span $\Delta(s) < 2 \cdot \mu(s)$ for almost all decreasing swarms and $\Delta(s) < 5 \cdot \mu(s)$ for almost all increasing swarms. This difference can be explained by flash-crowd arrivals in some new and very popular swarms which lead to a large increase in peer populations. In summary, we conclude that already in a time frame of 36 h, which is rather small compared to the lifetime of a swarm, the swarm populations can vary heavily. It is important to keep that in mind if parameter settings of locality aware
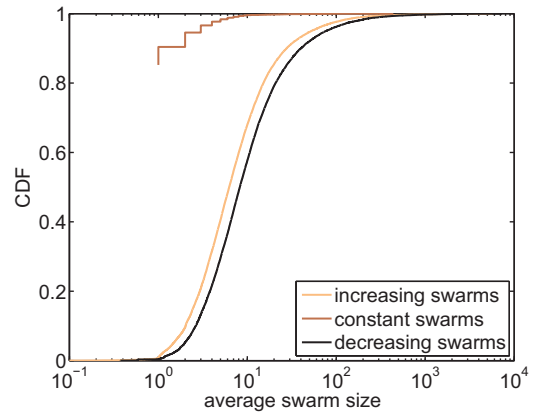

Fig. 3. CDF of the swarm sizes for increasing, constant, and decreasing swarms (measurement setup `TV.`).
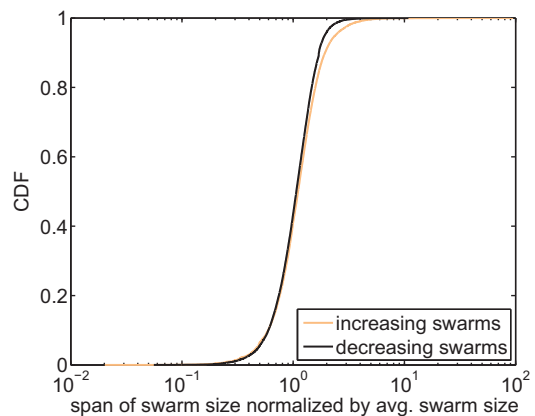

Fig. 4. CDF of the span of swarm sizes for measurement experiment `TV.`.

mechanisms need to be adjusted based on current swarm populations.

Next, we study how the time dynamics correlate with the swarm sizes, i.e., whether large swarms are subject to large variations or not. To this end, we calculate the coefficient of correlation ρ of the average swarm size to three values representing the variation: the span $\Delta(s)$, the standard deviation $\sigma(s)$ and the coefficient of variation $c(s)$ (cf. Table 3). We observe that the span $\Delta(s)$ and the standard deviation $\sigma(s)$ is strongly correlated to the average swarm size for increasing and decreasing swarms ($\rho > 0.65$). However, these correlations vanish if we take the coefficient of variation $c(s)$ instead of the standard deviation $\sigma(s)$. That means that larger swarms tend to have larger variations of the swarm population which is not very surprising. However, the variation normalized by the average size $\mu(s)$, i.e., the relative change in the swarm

Table 3
Coefficients of correlation ρ of the average swarm size $\mu(s)$ and the variation ($\Delta(s)$, $\sigma(s)$, and $c(s)$) for increasing and decreasing swarms.

	$\rho(\mu(s), \Delta(s))$	$\rho(\mu(s), \sigma(s))$	$\rho(\mu(s), c(s))$
Increasing swarms	0.694521	0.667706	−0.038363
Decreasing swarms	0.671585	0.653354	−0.059542

population is not correlated with swarm size. Hence, large swarms do not grow or shrink disproportionally fast.

Finally, we illustrate the correlation between the average swarm size $\mu(s)$ and the coefficient of variation $c(s)$ of the swarm size with a scatter plot in Fig. 5 for the swarms of the TV. data set, sorted by swarm size. The coefficient of variation $c(s)$ for most of the swarms s is between 0 and 1, on average it is 0.2795. In addition, we observe a set of swarms (around 1% of the measured swarms) where $c(s)$ is very close to 1. The reason for this band are frequent jumps of the swarm size (reported at the PirateBay website) between 0 and the actual swarm size which we attribute to an error in this website. However, this should have only a minor impact on our results since only 1% of the TV. data set shows this behavior and the TV. data set is the only one we measured by parsing the website (cf. Table 1). In order to show that the peculiar shape of the scatter plot is not owed to chance, we present a short mathematical derivation for the theoretical minimum of the coefficient of variation $c(s)$. Since we capture $R = 96$ samples of the size of a swarm s for the TV. experiment, the minimum standard deviation $\sigma(s)$ for a given average swarm size $\mu(s) \in [a; a + 1[$ is obtained when we measure k times a size of a and $R - k$ times a size of $a + 1$ (for $a \in \mathbb{N}$). Thus, it is

$$\mu(s) = \frac{ka + (R-k)(a+1)}{R} \quad \text{and} \quad \sigma(s) = \sqrt{\frac{ka^2 + (R-k)(a+1)^2}{R} - \mu(s)^2} = \frac{1}{R}\sqrt{(R-k)k}$$

which explains the shape of the theoretical minimum for the measurements.

4.2.2. Diurnal fluctuations

Now we take a closer look at the fluctuations. The evolution of the size of four example swarms, which are taken from the set summarized in Table 5, is depicted in Fig. 6. The selection of these swarms allows us to show principal differences between swarms even if they share the same type of content. Here, the swarm population over time is shown, with the base unit of the y-axis being 10^3 peers.

We observe that there are variations in the population of each swarm, as well as quantitative and qualitative differences in these variations between the swarms. While swarm (D), which is sharing a movie in English, shows only small changes in its peer population, the size of swarm (C) exhibits a periodic behavior. We attribute this to the fact
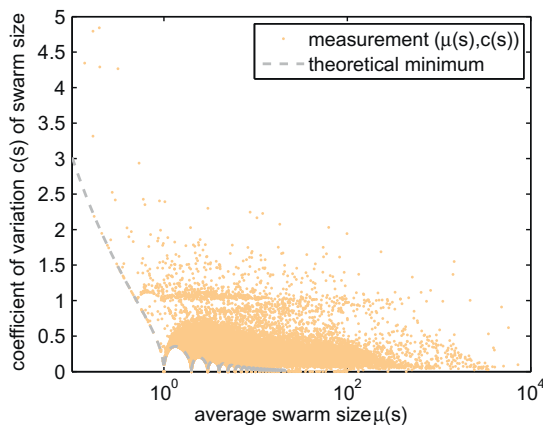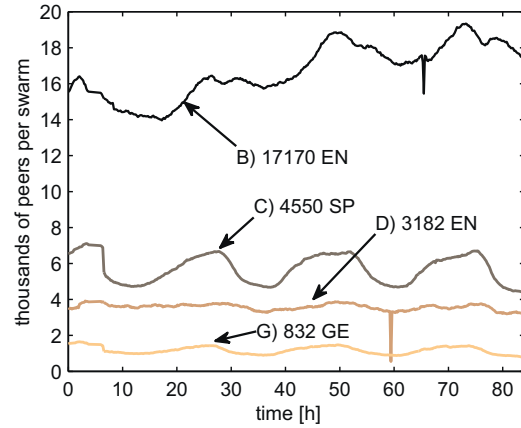

Fig. 6. Total swarm size of exemplary swarms (measurement setup Grp.) as defined in Table 5.

that in this swarm, a movie in Spanish is distributed. In order to check how many peers of that swarm are located in Spain we use the GeoIP service of MaxMind [26] to map the IP addresses to countries. In fact, more than 94% of the peers are from Spain and only about 2% from South America. Therefore, the swarm population increases during the daytime in this region these regions and decreases again afterwards. Swarm (G), sharing a German movie, shows a similar characteristic. The fluctuations are not as clearly visible as for swarm (C), but in relation to the average swarm size, the population of swarm (G) fluctuates to roughly the same degree as swarm (C).

The development of the peer population of swarm (B) is a superposition of a continually increasing popularity and a 24 h cycle like for swarms (C) and (G). While swarm (D) distributes content that seems not to be preferred regionally, the movie shared in swarm (B) seems to be more popular in a specific part of the world.

We now want to determine the amount of swarms that show a diurnal behavior similar to swarms (B), (C) and (G), in order to judge the relevance of this effect for the performance evaluation of locality awareness mechanisms. To that end, we use a method called periodicity transform which automatically detects periodicities for a given data set. In particular, we rely on the 'M-best' algorithm as introduced in [25] that returns a list of the $M = 10$ best periodicities. From the M best periodicities that are $\{\tau_i : 1 \leqslant i \leqslant M\}$, we calculate the autocorrelation ρ_i at lag τ_i and select the best period of duration τ_k with maximum, positive autocorrelation ρ_k, i.e. $k = \arg(\max\{\rho_i : 1 \leqslant i \leqslant M\})$. We also tried the other methods described in [25], but the M-best algorithm delivered the best results in finding periodicities of around 24 h.

Fig. 7 shows the CDF of the length of the 'best' period for the number of seeders, the number of leechers, and the entire swarm size for the TV. data set. It can be seen that the three different curves show a similar behavior. In particular, the curves for the number of leechers and seeders are almost identical, showing that the leechers mainly determine the diurnal behavior. Furthermore, we observe that roughly for 60% of the swarms the 'best' period is between 21 h and 27 h. There is no discontinuity in the CDF at 24 h

Fig. 5. Coefficient of variation c_i of the size of swarm i vs. average size of swarm i for measurement experiment TV.

since the M-best analysis is not able to completely ignore all other effects changing peer populations such as increasing or decreasing popularity of the content or flash-crowd arrivals.

Fig. 8 shows the autocorrelation ρ_k to the best period of duration τ_k. Again, the three different curves are quite similar. We observe that from the swarms in the TV. data set only 8.36% show a strong correlation $\rho_k > 0.7$. As a summary of the time-dynamics analysis, we see that for roughly 5.7% of the swarms a day–night behavior can be observed. To be more precise, for these swarms the autocorrelation is larger than 0.7 for the best period, while the duration of the period is about 1 day, i.e. between 21 h and 27 h.

4.3. Distribution of peers over AS's

One important performance indicator for locality aware mechanisms, typically used in related studies [4–6,8], is the amount of inter-domain traffic which can be saved by its application. In such investigations, the distribution of peers over AS's can play a major role for the potential savings [6,8]. As a consequence, we consider in this section statistics on the number of AS's which contain peers participating in the same swarm and on the average number of peers located in one AS. For this purpose, we use the Mov. and Mus. data sets since they contain a large number of swarms together with the IP addresses of the peers so that we can map them to AS's. The distribution of peers over AS's of swarms sharing regional content (Reg. data set) is presented in Section 4.7.

We present the CDFs for the average number of peers per AS for swarms of the Mov. data set in Fig. 9. Note that the x-axis is scaled logarithmically. The swarms are grouped according to their average size as shown in Table 4 together with the relative size of each group. We observe that for an increasing mean swarm size, the average number of peers per AS grows. However, this value is still small even for the largest swarms. This is in line with literature [6,7,18,19]. Considering the Mus. data set leads to the same conclusions. In fact, the average number of peers per AS is even smaller for these swarms. The concrete numbers
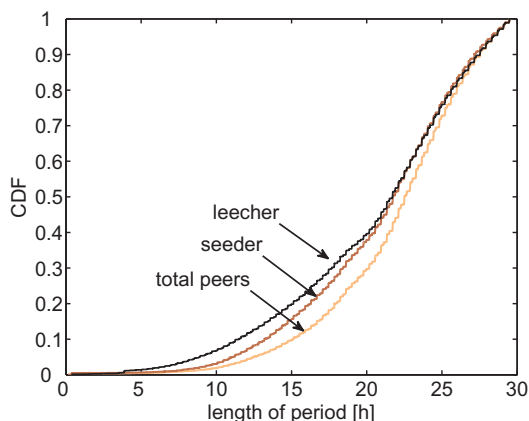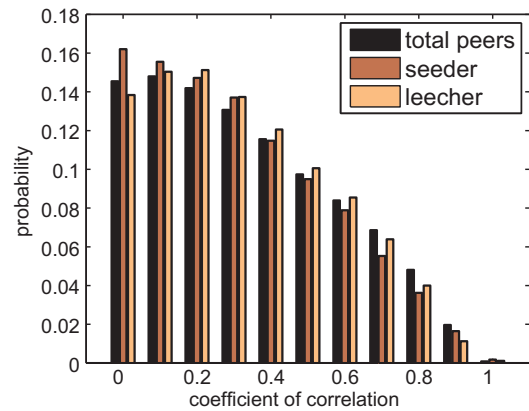

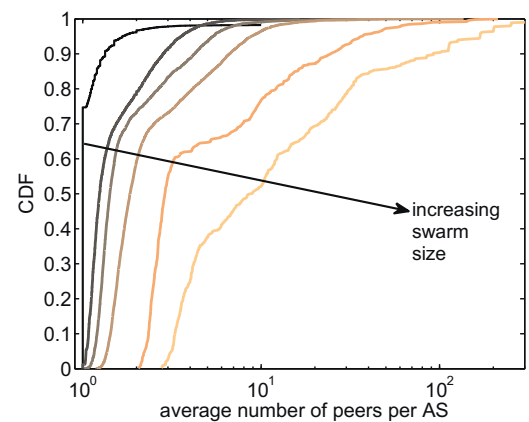**Fig. 8.** Autocorrelation to the best period for TV.

Fig. 9. CDF of average number of peers per observed AS. Swarms (Mov.) are grouped according to their size, cf. Table 4.

corresponding to Table 4 and Fig. 9 can be found in our technical report [12]. In Section 4.6, where we analyze the distribution of peers over countries, we show CDFs also for the maximum number of peers per AS (cf. Fig. 16).

Another important characteristic of a swarm is the absolute number of AS's because swarms that are distributed over fewer AS's but with more peers per AS can likely utilize locality promotion mechanisms more efficiently. To this end, we consider the movie files (Mov.) as well as the music files (Mus.). Fig. 10 shows the CDF of the number of AS's per swarm for both data sets. Since there are more peers involved in swarms offering movie contents, there are also more different AS's involved than in swarms providing music files. On average, there are 65% more AS's involved in movie swarms than in music swarms. In particular, if the CDF of the number of AS's for movie swarms is normalized by a factor of 1.65, it is nearly identical to the CDF for music swarms. The maximum number

Fig. 7. Length of period for TV. by calculating the periodicity transform using the M-Best Algorithm [25].

Table 4
Percentage of swarms grouped according to their size for movie files (Mov.).

[0; 25[	[25; 50[	[50; 100[	[100; 500[	[500; 1e3[	[1e3; ∞[
0.8580	0.0703	0.0294	0.0347	0.0040	0.0036

of observed AS's is 1744 for movie swarms and 809 for music swarms, respectively. We will explore the distribution of peers over AS's in more depth in Section 5 where we provide a model for the probability that a peer belongs to a certain AS.

4.4. AS clustering of peers

A fundamental pre-condition of keeping BitTorrent traffic within a given AS is that several peers sharing the same file are present in that AS. Therefore, we study in this section which fraction of swarms actually have the possibility to exchange data with local neighbors. To that end, we count the number of peers in every swarm which are located in an AS with at least α peers of the swarm. To obtain the AS clustering of peers δ_α of a swarm, we normalize this number by the swarm size. In other words, δ_α represents the fraction of peers in the swarm having at least $\alpha - 1$ other peers of the same swarm in their AS. In Fig. 11 we show CDFs of the AS clustering for $\alpha \in \{3, 4, 5\}$. We observe that in roughly 89% of the Mus. swarms no AS exists where at least 3 peers are present ($\delta_3 = 0$) and only in about 3.5% of the swarms the majority of peers ($\delta_3 = 0.5$) can be clustered in their AS's. Considering the movie files (Mov.), the probability to find clusters of peers within an AS is higher since these swarms are larger. Still, in about 88% of the movie swarms there is no AS with at least 5 peers. Thus, locality awareness will only be useful in a rather small fraction of the swarms. However, this statement does not refer to the question about which fraction of the total BitTorrent *traffic* can be influenced by locality awareness.

To answer this question, we first study the total amount of traffic produced by the swarms in the Mus. and Mov. data set. For that purpose, we take the number of peers in a swarm as an indicator of how much traffic a swarm produces in relation the other swarms and assume that peer access capacities are not correlated with the swarm sizes. Therefore, they can be neglected in our simple approximation. Fig. 12 presents the cumulative estimates ('music traffic T_0', 'movie traffic T_0') for the top x% of the largest swarms normalized by the total amount of traffic. The figure reveals that 10% of the swarms of the Mov. data set contain 80% of the peers and are consequently responsible for
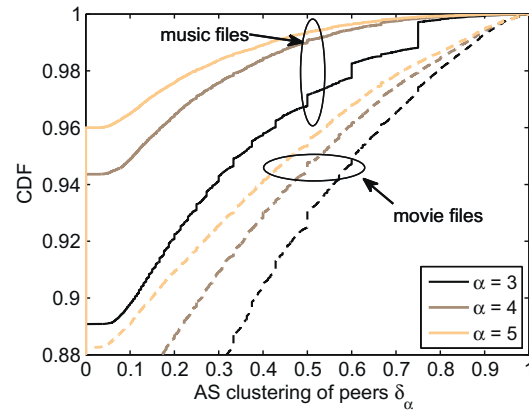

Fig. 11. CDF of AS clustering of peers δ_α for music (Mus.) and movie (Mov.) files.

the same fraction of the total traffic according to the aforementioned assumptions. If we weight the number of peers in a swarm with the size of the exchanged file ('traffic T_f', legend: 'with file sizes') to estimate the amount of traffic, we obtain almost the same results as for taking just the number of peers ('w/o file sizes'). This is in particular true for the movie traffic. For the music files the difference is small.

Next, we develop a very simple and optimistic approximation for the potential of locality awareness. This approximation is based on the results for the AS clustering δ_α. For each swarm, we calculate δ_2, i.e., the fraction of peers in the swarm which are not the sole peer in their AS. We assume an ideal locality algorithm which achieves that those peers produce no inter-domain traffic and neglect which peers are seeders and leechers and possible performance degradations for simplicity reasons. Then, δ_2 is the fraction of 'potentially local traffic' of that swarm. Fig. 12 shows this value weighted by the total traffic of the swarm for the music and movie files. The figure shows that it has almost no impact on this approximation whether we take into account the file sizes ('L_f') or not ('L_0') for the calculation of the total traffic a swarm produces. Furthermore, the figure confirms our finding that
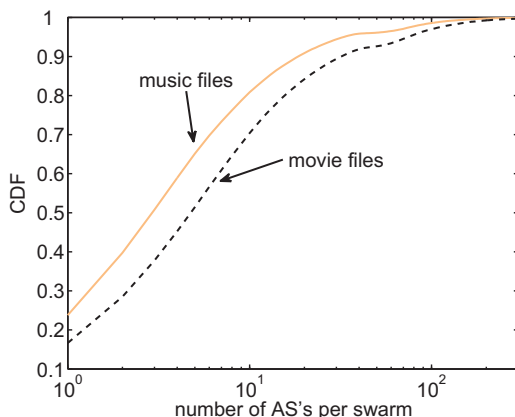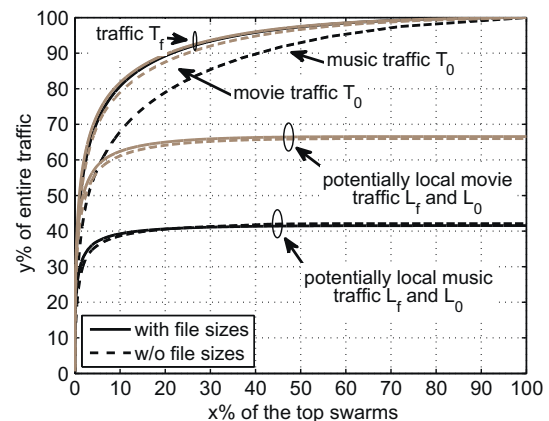

Fig. 10. Number of AS's per swarm (Mus., Mov.).

Fig. 12. Total and potentially local traffic of the top x% of the swarms (Mus., Mov.) with and w/o considering the size of the exchanged files.

locality awareness is only useful in a small subset of all swarms. However, it shows in addition that the potential savings of inter-domain traffic are quite larger in the big swarms which are responsible for the vast majority of Bit-Torrent traffic. Therefore, the overall optimization potential of locality awareness is about 65% for the movie files (`Mov.`) and roughly 40% for the music files (`Mus.`). In other words, around 35% (60%) of the overall movie (music) traffic is produced by peers which are the only one in the AS. Therefore, no locality awareness mechanism can avoid this inter-domain traffic. In summary, we conclude from this section that the overall optimization potential for locality awareness is large even if the mechanisms will only be useful in the top 20% of the swarms.

4.5. Relation of the number of peers and the AS degree

In this section we investigate to which degree the size of an AS is correlated with the number of peers it contains. For that purpose, we study two metrics representing the "size" of an AS: the AS rank and the AS degree. Both metrics are provided by CAIDA [27]. The AS degree is defined as the number of AS's to which a given AS is connected. Like in [28] we use the AS degree as an indicator for the size of the AS. To obtain the AS rank of a given AS, CAIDA basically orders all AS's according to their size and defines the AS rank of a given AS as its index in this ordered list. For this investigation we use the `Mus.` and `Mov.` data set since these contain large numbers of swarms and their distribution of peers over AS's.

First, we check the correlation of the total number of peers per AS to the size of the AS. To this end, we calculate the total number of peers in a given AS as the sum of the number of peers in this AS of all swarms in the data set. Then, we correlate the total number of peers per AS with the AS degree and the AS rank obtained from CAIDA. This calculation shows that the total number of peers in an AS is neither correlated to the AS rank nor to the AS degree. The concrete values for the correlation to the AS rank are -0.0962 and -0.0834 for the `Mus.` and `Mov.` data set, respectively. The corresponding values for the correlation to the AS degree are 0.1492 (`Mus.`) and 0.1020 (`Mov.`).

Next, we calculate the correlation of the number of peers per AS with the corresponding AS degree for each swarm. That means, we get one correlation coefficient for every swarm in the data set and plot CDFs of this value for the 100 and 10,000 largest swarms (cf. Fig. 13). Although some swarms exist in the top 10,000 swarms of both data sets where the correlation is high, most of the swarms do not have this strong correlation. In particular, these swarms are not among the 100 largest swarms. Therefore, we conclude that within a given swarm it is quite unlikely that the number of peers per AS is correlated with the AS degree. A possible explanation for that rather unexpected result is that there is a large number of AS's in every swarm which contain only 1 or 2 peers. Still, these AS's may have a high AS degree which leads to low values for the correlation.

To avoid this influence of the large number of AS's with only a few peers, we now focus on the top AS of every swarm. In this way, we limit our investigation to those
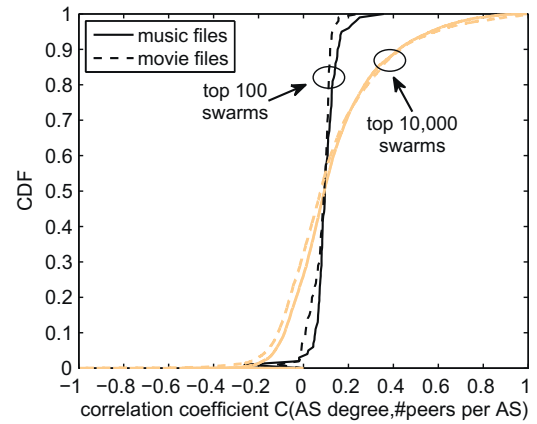

Fig. 13. CDF of the correlation of the AS degree and the number of peers per AS for every swarm (`Mus.`,`Mov.`).

AS's with a large number of peers. In Fig. 14 we calculate the number of peers in the top AS's of the x largest swarms and correlate these x numbers to the corresponding AS degree and AS rank. We observe that the correlation with the AS degree is stronger than the one with the AS rank. Furthermore, the correlation decreases when we increase x, i.e., when we take into account more swarms. In particular, the correlation of the AS degree and the number of peers in the top AS of the 100 largest swarms (`Mus.`) is close to 1. That means, for the AS's where the number of peers is large, this number is correlated to the AS degree.

4.6. Another metric for locality: country codes

While AS affiliations are a popular metric describing which peers are nearby, other metrics such as the number of IP- or AS-hops, similarity of CDN redirection behavior [17], or geographic proximity can also be used for that purpose. In this section we investigate in which way the results of the previous section are affected if we use another criterion than the AS affiliation. For feasibility reasons we select the geographic proximity out of the aforementioned example metrics and map every IP address to a country code using the MaxMind GeoIP service [26].
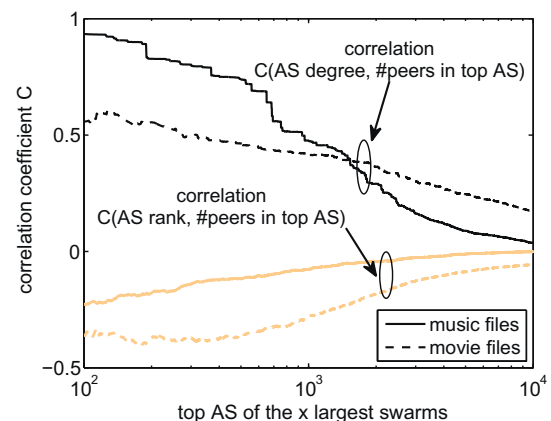

Fig. 14. Correlation of the AS degree and AS rank with the number of peers in the top AS of each swarm (`Mus.`,`Mov.`).

First, we compare the number of peers per AS to the number of peers per country. For that purpose, we calculate the average number of peers per AS and per country for every swarm (`Mus.` and `Mov.`) and show CDFs over all swarms in Fig. 15. We observe that the number of peers per country is higher than per AS. For the `Mus.` data set the mean number of peers per country (averaged over all swarms) is about 2.3 times higher than the mean number of peers per AS. For the `Mov.` data set the same relation is about 6.2. This seems reasonable since most countries contain several AS's. Fig. 16 is similar to Fig. 15 but presents the maximum number of peers per AS and per country instead of the average numbers. That means that we select from each swarm that AS and that country with the highest number of peers. Again, we observe that the number of peers per AS is lower than per country. Second, we investigate the number of countries per swarm in analogy to Fig. 10, which is based on the AS affiliations. The corresponding figure for the country codes is very similar to Fig. 10, and we therefore omit it. The only difference is the one already observed in Fig. 15 that there are on average more peers per country than per AS.

Hence, using country codes instead of AS affiliations leads to coarser classification of peers and consequently higher numbers of peers in the same class. Therefore, keeping traffic local in a given country should be easier since it is more likely to find local neighbors than in the same AS.

4.7. Characteristics of regional swarms

We have already seen the effect regional content has on the evolution of the swarm size over time. We now take a closer look at the topological characteristics of swarms sharing this content. These swarms are contained in the data sets `Reg.` and `Grp.` The `Grp.` data set comprises 16 example swarms of different average sizes distributing movies in German, Spanish, Chinese or English (cf. Table 5). For these swarms, we analyze the number of AS's and the top AS fraction of the swarm, i.e., the maximum number of peers in an AS of that swarm normalized by the swarm size (cf. Fig. 17). In this figure, the swarm size (given in Table 5) is indicated by different colors on a
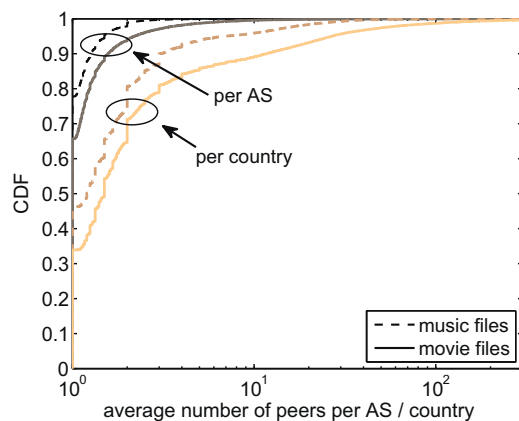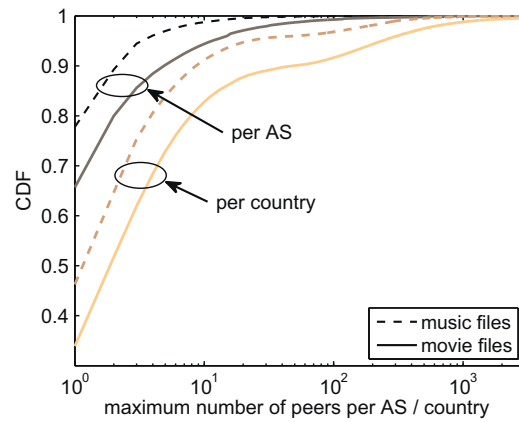

Fig. 16. CDF of the maximum number of peers in a country per swarm (`Mus.`,`Mov.`).

logarithmic scale. Swarms sharing regional content have a high top AS fraction (20% to 50%) and are spread over comparably few AS's. In contrast, swarms sharing internationally interesting content, i.e., in English, have a small top AS fraction (below 10%) and are spread over more AS's.

Swarm D is an exception here. We have seen in Section 4.2 that the peer population within swarm D remains almost constant over time and does not show any periodic day–night pattern. Thus, the swarm distributes content that seems not to be preferred regionally. However, swarm D shows the highest skewness in terms of number of peers per AS compared to the other swarms. In particular, 30% of the peers belong to the same AS with the AS number 30058. A closer look reveals that the company responsible for this AS offers its customers to rent dedicated or virtual servers located in this AS. This permits a single customer to run a large number of peers on different virtual nodes which could be used to insert fake peers in the swarm in order to disturb the distribution process. This might be an explanation of the high fraction of peers in swarm D in AS 30058.

Next, we move from the `Grp.` data sets with 16 example swarms to the `Reg.` data sets containing 120 swarms exchanging regional movies observed at the index server PirateBay.org in May 2009. This set is more suitable for statistical analysis since the number of swarms is higher and the swarms are not selected by hand as it is the case for `Grp.` The fact that users are interested in regional content leads to a high top AS fraction, which is the relative number of peers in a swarm's top AS. This is especially true for Spanish content, see Fig. 18. Here, the top AS of each swarm in the `Reg.` set is used for comparison, i.e., the AS containing most peers from a swarm. In this graph, a CDF of the relative share of peers that are located in these

Fig. 15. CDF of average number of peers per observed AS and per country for the `Mov.` and `Mus.` data sets.

Table 5

Individually measured swarms over time (`Grp.`) using the following notion: *(ID) average swarm size* & *language.*

(A) 21,351 EN	(B) 17,170 EN	(C) 4550 SP	(D) 3182 EN
(E) 1390 SP	(F) 972 GE	(G) 832 GE	(H) 626 GE
(I) 579 SP	(J) 479 EN	(K) 473 GE	(L) 351 GE
(M) 289 GE	(N) 258 EN	(O) 217 SP	(P) 81 CN
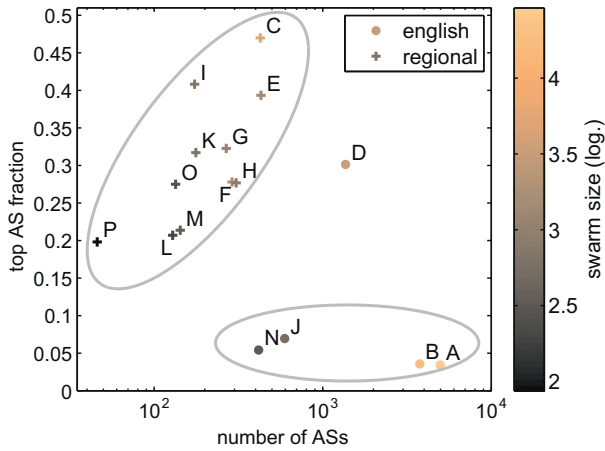
Fig. 17. Scatter plot of the number of AS's and the top AS fraction (`Grp.` data set, cf. Table 5).

Fig. 19. Relative number of peers in *i*th top AS (`Reg.`).

AS's is plotted for swarms with Dutch, French, Italian and Spanish content.

While in all cases there are at least 10% of the total swarm population in the top AS, this share is between 40% and 48% for the Spanish content, implying a high degree of peer grouping. To judge whether this phenomenon only exists for a single AS, we evaluated also the second to fifth largest AS's of the swarms in the `Reg.` data set, cf. Fig. 19. It appears that the top AS of a swarm contains significantly more peers than the other AS's, although these are still holding around 5% of the total swarm population.

We affirm this result by comparing the kurtosis, i.e., the fourth moment of a distribution that indicates statistical peaks, of the number of peers per AS for the swarms in the `Reg.`, the `Mus.` and `Mov.` sets. The results are shown in form of a CDF in Fig. 20.

The regional swarms show a much higher kurtosis than the two larger and more general sets. This leads us to the conclusion that the concentration of a larger fraction of the swarm in the same AS is much more common in regional swarms. This means that the regional interest in a shared file can play a significant role in the suitability of the according swarm for locality promotion, something previously underestimated. In particular, the high kurtosis
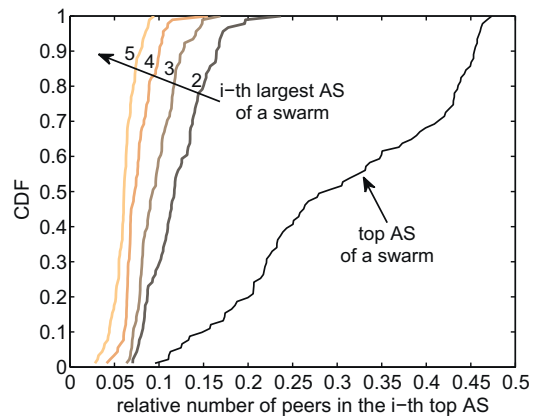
values for a certain fraction of swarms providing music or movie files in Fig. 20 indicates that this phenomenon of regional interests with many peers in the top AS can be observed for any kind of content.

4.8. Measurement summary

From the results presented above, we make the following main observations for the characterization of BitTorrent swarms and their distribution in the Internet.

Considering the swarm statistics according to the offered content (i.e., TV shows, movies, and music) we observe that the larger the offered content is in terms of volume, the larger the average and maximum number of peers is in such a swarm, as already shown in less detail in [19]. Additionally, our results show that the distribution of peers among the swarms follows the Pareto principle for the different measurement sets (1), (4) and (5) which contain random files. This means that 80% of all peers belong roughly to the top 20% swarms for all media types. The Pareto principle cannot be observed for measurement set (2), (3), and (6) since we only consider popular or recently published content there. These recently published torrents are highly popular. This is reasonable since users are typically interested in new contents, recently broadcasted movies etc. We studied the distribution of peers over AS's of the
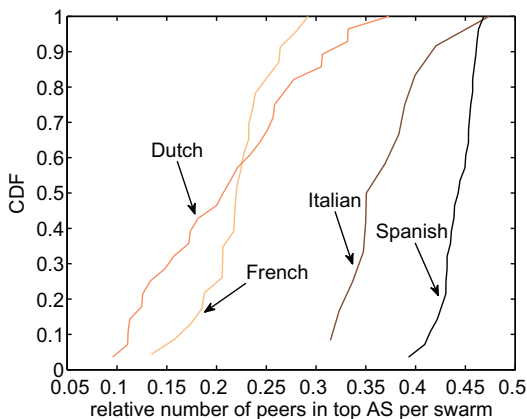

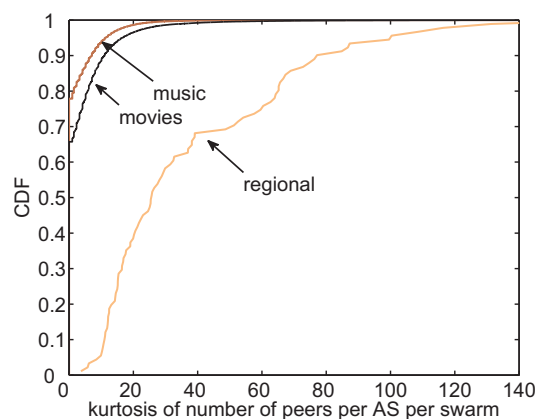**Fig. 18.** Relative number of peers in a swarm's top AS (`Reg.`).

Fig. 20. Kurtosis of number of peers per AS (`Reg.`,`Mus.`,`Mov.`).

swarms and showed that the average number of peers per AS is small for most of the swarms. However, the distribution of peers over AS's is skewed so that a high fraction of the peers is contained in the few top AS's of the swarm. Previous studies, e.g., [6,8], revealed that this can have a strong impact on the performance of traffic optimizations schemes, especially for swarms sharing regional content, where the skewness in the peer distribution is higher. Hence, quantitative characterizations (cf. Section 5) of the distribution of peers over AS's are required for a meaningful performance evaluation of traffic optimization schemes. In addition, our measurements show that the fraction of swarms with AS's where more than 5 peers are located in at least one AS is quite small. Nevertheless, the optimization potential of locality aware mechanisms remains high since peers in the large swarms, which produce the majority of the traffic, can be clustered in their AS. As a consequence, it would be an option to concentrate traffic optimization efforts on the relatively low number of swarms with larger content and high popularity because the potential gains are much higher than for small swarms. Not only does a larger content lead to more traffic, but also the possibilities for locality promotion are more numerous in larger swarms, where there are more peers in one AS in general.

Also, especially for regional content we observe a day–night behavior of the swarm size since mostly users of a certain region (within a similar time zone) are interested in that content, e.g., movies in French are mostly downloaded by users from France. In general, we found for 5% of the investigated swarms a clear statistical indication for day–night behavior. Therefore, the efficiency of traffic optimization schemes may vary over time. Also, a one-time observation of a swarm may not suffice to characterize it for its suitability for locality promotion, even if it is no longer in its flash-crowd phase. When the classification of peers is done on the basis of the country code instead of the AS affiliation, we observe that more peers are in the same class and therefore it is easier to keep traffic within that class of peers. Finally, the measurements reveal that for a very small number of swarms (which are not the large ones) the number of peers in an AS is correlated to the AS degree.

4.9. Limitations of the measurement study

There are some limitations of our measurement study which we present here so these can be taken into account when using our results. First, we studied only swarms which use a tracker to request an initial set of peers and no tracker-less swarms. Second, our measurements rely on the assumption that the information obtained from the websites and the trackers are correct. Furthermore, we did not try to contact the peers we received from the trackers. Therefore, it is possible that some company inserted fake peers in order to disturb the distribution progress which would result in a smaller number of peers actively participating in a swarm than the one we measured. Third, we used different measurement methodologies for different data sets because we refined our methodology during the course of this work. This has

two consequences: (1) not all types of data are available for all data sets (namely distribution of peers over AS's and measurements over time) and (2) the results might be influenced by the used measurement methodology. Overall, we argue that these limitations have only a minor impact on the presented results. To support this we cross-checked the results using all data sets for which the corresponding type of measurement was available, provided explanations of differing results, and compared our results to the ones described in literature.

5. Statistical characterizations of BitTorrent swarms

Based on the measurements presented in Section 4 we develop a set of characterizations for BitTorrent swarms which can be used for performance evaluations of locality awareness solutions for BitTorrent. Namely, we model the distribution of peers of a single swarm over AS's and fit the swarm population, the number of AS's over which a swarm is distributed, the fraction of the swarm located in the top AS, and the size of the shared file with stochastic distributions for the data sets `Mus.`, `Mov.`, and `Reg`. Finally, we present the correlation of these values as multivariate correlation matrices.

5.1. Power-law of the distribution of peers over AS's

As we have seen from the measurement results presented in Section 4, one key aspect for modelling BitTorrent swarms is the skewed peer distribution. In this section, we present a simple model which returns the probability $P(k)$ that a peer belongs to the kth largest AS within a swarm consisting of n different AS's. In particular, we investigate whether the peer distribution among the different AS's follows a power-law, which means

$$P(k) = a/k^b + c. \tag{3}$$

Therefore, we consider all swarms $\mathfrak{I}_n$ consisting of exactly n different AS's from `Mus.` and the `Mov.` data set, respectively. For each swarm $i \in \mathfrak{I}_n$, we measure the ratio $\widetilde{P}_i(k)$ of peers belonging to the kth largest AS in swarm i for $k = 1, 2, \ldots, n$. Then, we compute the average ratio $\widetilde{P}(k)$ over all swarms, yielding at

$$\widetilde{P}(k) = \frac{1}{|\mathfrak{I}_n|} \sum_{i \in \mathfrak{I}_n} \widetilde{P}_i(k). \tag{4}$$

Fig. 21 shows the measured ratio $\widetilde{P}_i(k)$ of peers belonging to the kth largest AS within a swarm consisting of $n = 40$ different AS's. All swarms consisting of exactly n different AS's are considered from the `Mus.` data set. The observed ratio $\widetilde{P}_i(k)$ is then compared with the power-law model function as defined in Eq. (3). The parameters a, b, c of the model function are retrieved by means of non-linear regression. We used the optimization toolbox of Matlab to find an optimal fitting function for the given measurement data. Optimal in this case means to find the unknown parameters a, b, c in Eq. (3), such that the mean squared error is minimized. As a result, we obtain $P(k) = 0.0769/k^{0.8013} + 0.0134$ which is plotted as solid curve. Fig. 21

indicates that the power-law describes quite well the peer distribution among AS's.

The goodness-of-fit for the model function $P(k)$ is expressed by means of the coefficient of determination R^2. A value close to one means a perfect match between the model function and the measured data. For the measurements given in Fig. 21 and the obtained model function, the coefficient of determination is $R^2 = 0.978035$ indicating the good match in a statistical way. In our case, the coefficient of determination can be computed as follows

$$R^2 = 1 - \frac{\sum_{k=1}^{n} \left(\widetilde{P}(k) - P(k) \right)^2}{\sum_{k=1}^{n} \left(\widetilde{P}(k) - 1/n \right)^2}. \qquad (5)$$

In the following, we have computed the optimal parameters of the power-law function as defined in Eq. (3) for all swarms consisting of exactly n different AS's. Again, the coefficient of determination R^2 is used to measure the goodness-of-fit. Fig. 22 shows a scatter plot of the number n of different AS's in a swarm vs. R^2 for the Mus. data set. The maximum number of observed AS's is 1744 for movie swarms and 809 for music swarms. As we can see, the match between the measurement data and the power-law model function is very good and the coefficient of determination is above 0.9. In [12], the power-law describing the distribution of peers over AS's of BitTorrent swarms was also shown for the Mov. data set. In order to provide a model for BitTorrent swarms, the file size, the size of a swarm, and the number of AS's per swarm is required in addition to the parameters of the power-law model. This is discussed in the following section.

5.2. Additional parameters of BitTorrent swarms

In order to provide input for the evaluation of locality awareness mechanisms under more realistic conditions, we introduce statistical characterizations for music files, movie files, and files of regional interest based on the measurements for the Mus., Mov., and Reg. data sets, respectively. The considered features of BitTorrent swarms relevant for traffic optimization comprise (a) the size of a
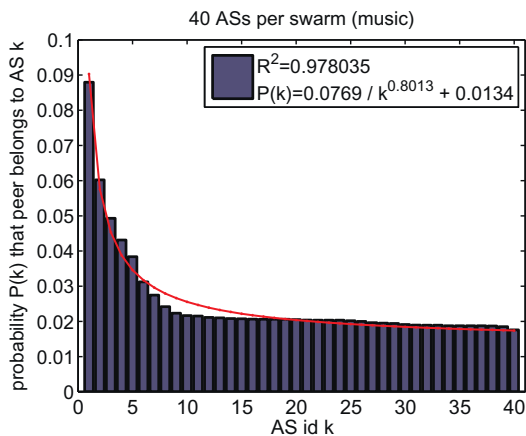

Fig. 21. Comparison of the measured ratio $\widetilde{P}(k)$ and the theoretical probability $P(k)$.
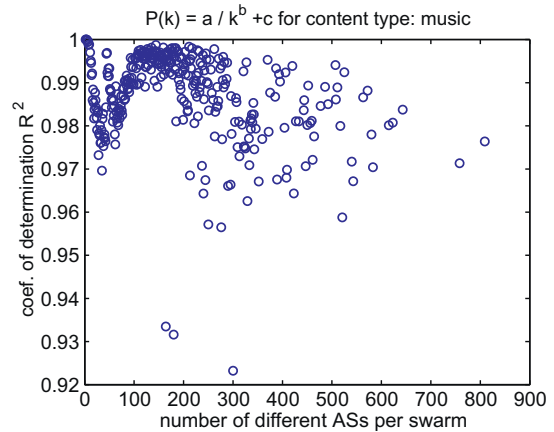

Fig. 22. Goodness-of-fit between the measurement data and the power-law model, cf. Eq. 5.

swarm, (b) the number of AS's per swarm, (c) the top AS fraction, and (d) the size of the provided file in the swarm.

Tables 6–8 show the distribution model of these features f, the mean value $\mu(f)$, the coefficient of variation $c(f)$, and the corresponding model parameters. For the Mov. and Mus. data sets, we excluded swarms with less than 10 peers from our consideration since most of the Bit-Torrent users (around 80%, cf. Fig. 2) do not belong to these swarms and locality awareness is expected to have only a very small impact in these swarms (cf. Section 4.4). The Reg. data set does not contain those small swarms and we therefore included all swarms from this set in the characterizations. Using the measurement data, the maximum likelihood estimates of the parameters for the different model distributions were calculated. The goodness-of-fit (gof) of the model distribution and the measurement data is expressed by the coefficient of determination R^2 which takes values from 0 to 1. A value of $R^2 = 1$ shows that the model function and the measurement data are identical. Thus, we can see a very good match between the measurement data and the model functions. An exception is the size of movie files (Mov.) and regional files which only have a gof of $R^2 = 0.86$ and $R^2 = 0.83$, respectively. This can be explained by the fact that the distributions of these file sizes show a strong peak. In particular, 45.85% of all movie files have a size between 650 MB and 750 MB which corresponds to the size of a regular compact disc. In addition, about 8.46% of the swarms have a file size between 1350 MB and 1450 MB. Fitting only the file sizes of the remaining 53.31% of the swarm gives significantly higher gof of 0.99 ('file size (impr.)' in Table 7). This is very similar for the Reg. data set. 51.65% of the swarms have a file size between 650 MB and 750 MB and 23.08% of them are between 1350 MB and 1450 MB. The number of the remaining swarms is too low to provide a meaningful fitting and we therefore suggest to use the corresponding values of the movie files.

However, as we have outlined in Section 4, there is a strong correlation between some of the features of BitTorrent swarms. Tables 9–11 show the multivariate correlation matrix for music, movie, and regional files, respectively. We observe that there is a strong correlation (>0.8 for Mus. and Mov., and > 0.6 for Reg.) between the

Table 6
Characterizations for music swarms with at least 10 peers.

Feature f	$\mu(f)$	$c(f)$	Model	Model parameter		R^2
Swarm size	46.15	2.66	Log-normal	$\mu = 3.18$	$\sigma = 0.89$	0.96
#ASs per swarm	28.31	1.39	Log-normal	$\mu = 2.97$	$\sigma = 0.74$	0.98
Top AS fraction	0.13	0.65	Log-normal	$\mu = -2.19$	$\sigma = 0.54$	1.00
File size	218.04	2.05	Log-normal	$\mu = 4.53$	$\sigma = 1.40$	0.97

Table 7
Characterizations for movie swarms with at least 10 peers.

Feature f	$\mu(f)$	$c(f)$	Model	Model parameter		R^2
Swarm size	85.34	5.64	Log-normal	$\mu = 3.42$	$\sigma = 1.06$	0.97
#ASs per swarm	33.67	1.95	Log-normal	$\mu = 3.01$	$\sigma = 0.86$	0.98
Top AS fraction	0.18	0.84	Log-normal	$\mu = -1.98$	$\sigma = 0.75$	1.00
File size	887.05	0.76	Gamma	$a = 1.91$	$b = 463.3$	0.86
File size (impr.)	975.74	0.97	Weibull	$\lambda = 985.97$	$k = 1.03$	0.99

Table 8
Characterizations for regional swarms with at least 1 peers.

Feature f	$\mu(f)$	$c(f)$	Model	Model parameter		R^2
Swarm size	1350.86	1.39	Log-normal	$\mu = 6.60$	$\sigma = 1.04$	0.92
#ASs per swarm	77.45	0.54	Gamma	$a = 3.58$	$b = 21.65$	1.00
Top AS fraction	0.31	0.38	Gamma	$a = 6.17$	$b = 0.05$	0.97
File size	1367.81	0.81	Log-normal	$\mu = 7.00$	$\sigma = 0.60$	0.83

number of peers in a swarm and the number of different AS's in a swarm.

In order to generate a random BitTorrent swarm based on this model, approximate methods for sampling correlated random variables from partially specified distributions can be used which are well known in literature, e.g. [29]. For these approximations, the information from the tables presented in the section can be used, respectively.

Table 9
Multivariate correlation matrix for music swarms with at least 10 peers.

	#Peers	#ASs	Top AS	File size
#Peers	1.0000	0.9100	−0.1364	−0.0048
#ASs	0.9100	1.0000	−0.2979	−0.0071
Top AS	−0.1364	−0.2979	1.0000	0.0129
File size	−0.0048	−0.0071	0.0129	1.0000

Table 10
Multivariate correlation matrix for movie swarms with at least 10 peers.

	#Peers	#ASs	Top AS	File size
#Peers	1.0000	0.8281	−0.0084	0.0043
#ASs	0.8281	1.0000	−0.2160	−0.0000
Top AS	−0.0084	−0.2160	1.0000	0.0086
File size	0.0043	−0.0000	0.0086	1.0000

Table 11
Multivariate correlation matrix for regional swarms with at least 1 peer.

	#Peers	#ASs	Top AS	File size
#Peers	1.0000	0.6102	0.5744	−0.0707
#ASs	0.6102	1.0000	0.0450	0.1259
Top AS	0.5744	0.0450	1.0000	−0.2670
File size	−0.0707	0.1259	−0.2670	1.0000

6. Conclusion

In this paper we measure and characterize real-life Bit-Torrent swarms. The results can serve as input for the design and assessment of traffic optimization techniques as currently discussed in the ALTO working group of the IETF. Still, they are of a general nature and therefore not limited to that purpose. A core part of our investigation is the result of a large-scale measurement campaign, where a comprehensive set of swarms has been investigated using a distributed tracker monitoring system. Measurements include swarm size distributions, ratio of seeder and leecher populations, time dynamics within a swarm, the distribution of peers over AS's and over countries of swarms, and characteristics of swarms with a certain content or region focus. We show that real-life BitTorrent swarm distributions are highly skewed and that this is in particular true for regional swarms. On the one hand, more than 90% of the observed AS's contain less than 10 peers and the average number of peers per AS is below 2 peers for 99% of the swarms with a very high variation leading to many single peer AS's. On the other hand, most of the peers (about 80%) belong to the top 20% of the swarms. Therefore, we argue that there is a large optimization potential for locality awareness since these large swarms are (1) responsible for the majority of the BitTorrent traffic and (2) especially suitable for locality aware mechanisms. For this reason, we have specified a simple AS swarm characterization for music, movie, and regional files provided in BitTorrent swarms which takes into account the swarm size, the number of different AS's per swarm, the top AS fraction, and the file size. These measurement results and the provided characterizations enable researchers to design algorithms as well as simulation studies and experiments for ALTO solutions based on real-world characteristics of BitTorrent swarms.

Acknowledgments

References

[1] Tobias Hoßfeld et al., An economic traffic management approach to enable the triplewin for users, ISPs, and overlay providers, in: Towards the Future Internet – A European Research Perspective, FIA Prague Book, IOS Press Books, ISBN 978-1-60750-007-0, 2009, p. 24.

[2] H. Xie, Y.R. Yang, A. Krishnamurthy, Y.G. Liu, A. Silberschatz, P4P: provider portal for applications, SIGCOMM Comput. Commun. Rev. 38 (4) (2008) 351–362.

[3] V. Aggarwal, A. Feldmann, C. Scheideler, Can ISPs and P2P users cooperate for improved performance?, SIGCOMM Comput. Commun. Rev. 37 (3) (2007) 29–40.

[4] R. Bindal et al., Improving traffic locality in BitTorrent via biased neighbor selection, in: Proc. IEEE International Conference on Distributed Computing Systems (ICDCS), Lisboa, Portugal, 2006.

[5] S. Oechsner, F. Lehrieder, T. Hoßfeld, F. Metzger, K. Pussep, D. Staehle, Pushing the performance of biased neighbor selection through biased unchoking, in: Proc. IEEE International Conference on Peer-to-Peer Computing (P2P), Seattle, USA, 2009.

[6] S.L. Blond, A. Legout, W. Dabbous, Pushing BitTorrent locality to the limit, Computer Networks 55 (3) (2011) 541–557.

[7] M. Piatek, H.V. Madhyastha, J.P. John, A. Krishnamurthy, T. Anderson, Pitfalls for ISP-friendly P2P design, in: Proc. ACM Workshop on Hot Topics in Networks (HotNets), 2009.

[8] F. Lehrieder, S. Oechsner, T. Hoßfeld, Z. Despotovic, W. Kellerer, M. Michel, Can P2P-users benefit from locality-awareness?, in: Proc. IEEE International Conference on Peer-to-Peer Computing (P2P), Delft, The Netherlands, 2010.

[9] F. Lehrieder, S. Oechsner, T. Hoßfeld, D. Staehle, Z. Despotovic, W. Kellerer, M. Michel, Mitigating unfairness in locality-aware peer-to-peer networks, Int. J. Netw. Manag. (IJNM), Special Issue on Economic Traffic Management 21 (2011).

[10] PlanetLab, An open platform for developing, deploying, and accessing planetary-scale services, 2009. <http://www.planet-lab.org/>.

[11] G-Lab Project, National platform for future internet studies, 2009. <http://www.german-lab.de/>.

[12] T. Hoßfeld, D. Hock, S. Oechsner, F. Lehrieder, Z. Despotovic, W. Kellerer, M. Michel, Measurement of BitTorrent Swarms and their AS topologies, Technical Report TR463, University of Würzburg, Würzburg, Germany, 2009.

[13] S. Khirman, Torrent as localisation – raw data, 2009. <http://www.khirman.com/blog/as_raw_data>.

[14] M. Izal, G. Urvoy-Keller, E.W. Biersack, P.A. Felber, A.A. Hamra, L. Garces-Erice, Dissecting BitTorrent: five months in a torrent's lifetime, in: Proc. Passive and Active Measurements (PAM), Antibes Juan-les-Pins, France, 2004.

[15] J. Pouwelse, P. Garbacki, D. Epema, H. Sips, The BitTorrent p2p file-sharing system: measurements and analysis, in: Proc. International Workshop on Peer-to-Peer Systems (IPTPS), Ithaca, NY, USA, 2005.

[16] L. Guo, S. Chen, Z. Xiao, E. Tan, X. Ding, X. Zhang, Measurements, analysis, and modeling of BitTorrent-like systems, in: Proc. ACM SIGCOMM Conference on Internet Measurement (IMC), USENIX Association, Berkeley, CA, USA, 2005, pp. 4–4.

[17] D.R. Choffnes, F.E. Bustamante, Taming the Torrent: a practical approach to reducing cross-ISP traffic in peer-to-peer systems, SIGCOMM Comput. Commun. Rev. 38 (4) (2008) 363–374.

[18] R. Cuevas, N. Laoutaris, X. Yang, G. Siganos, P. Rodriguez, Deep diving into BitTorrent locality (CoRR abs/0907.3874), 2009.

[19] H. Wang, J. Liu, X. Ke, On the locality of BitTorrent-based video file swarming, in: Proc. International Workshop on Peer-to-Peer Systems (IPTPS), 2009.

[20] RIPE NCC, RISwhois. <http://www.ripe.net/projects/ris/tools/riswhois.html>.

[21] A. Loewenstern, Dht protocol. <http://www.bittorrent.org/beps/bep_0005.html>.

[22] J. Kobza, S. Jacobson, D. Vaughan, A survey of the coupon collector's problem with random sample sizes, Method. Comput. Appl. Probab. 9 (4) (2007) 573–584.

[23] A. Binzenhöfer, T. Hoßfeld, Warum Panini Fußballalben auch Informatikern Spaß machen, in: H.-G. Weigand (Ed.), Fußball eine Wissenschaft für sich, Verlag Königshausen & Neumann, 2006.

[24] D. Qiu, R. Srikant, Modeling and performance analysis of BitTorrent-like peer-to-peer networks, SIGCOMM Comput. Commun. Rev. 34 (4) (2004) 367–378.

[25] W.A. Sethares, T.W. Staley, Periodicity transforms, IEEE Trans. Signal Process, 1999.

[26] MaxMind, Geolocation service. <http://www.maxmind.com/>.

[27] The Cooperative Association for Internet Data Analysis (CAIDA), AS Ranking, 2010. <http://as-rank.caida.org/>.

[28] L. Gao, On inferring autonomous system relationships in the Internet, IEEE/ACM Trans. Netw. 9 (6) (2001) 733–745. <http://dx.doi.org/10.1109/90.974527>.

[29] P.M. Lurie, M.S. Goldberg, An approximate method for sampling correlated random variables from partially-specified distributions, Manage. Sci. 44 (2) (1998) 203–218.

Tobias Hoßfeld studied computer science and mathematics at the University of Würzburg, Germany. He finished his PhD on performance evaluation of future Internet applications and emerging user behavior in 2009. Currently, he is heading the FIA research group "Future Internet Applications & Overlays" at the Chair of Communication Networks in Würzburg. His main research interests cover network virtualization, social networks, self-organization mechanisms in overlay networks and P2P systems, with a special focus on mobile environments like 3G and 4G, as well as investigations on Quality of Experience (QoE) for Internet applications like Skype, YouTube, or Web Browsing in general.
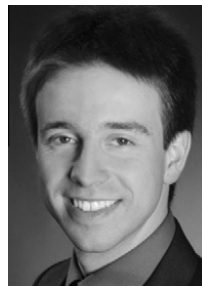
Frank Lehrieder studied computer science and business administration at the University of Würzburg/Germany and University Antonio de Nebrija in Madrid/Spain. In May 2008, he received his Master degree in computer science from University of Würzburg. Now, he is a researcher at the Chair of Communication Networks in Würzburg and pursuing his PhD. His special interests are performance evaluation and resource management for peer-to-peer networks and Future Internet infrastructures.

David Hock studied computer science and mathematics at the University of Würzburg/Germany and at the BTH in Karlskrona/Sweden. He received his diploma degree in computer science in spring 2009. Since then he has been pursuing his PhD as a research assistant at the Chair of Communication Networks in Würzburg. His current research focuses on wireless mesh networks and resilient IP networks.

Simon Oechsner studied computer science and received his Master degree from the University of Würzburg, Germany in 2004. He finished his PhD studies on "Performance Challenges and Optimization Potential of Peer-to-Peer Overlay Technologies" in 2010 under Prof. Tran-Gia at the Chair of Communication Networks at the same University. His research focuses on analytical and simulative performance evaluation of overlays. His interests include search and content distribution overlays, covering DHTs as well as currently popular file-sharing and video-streaming networks.

Zoran Despotovic is a Manager at NTT DOCOMO Euro-Labs. His research focus is mobile network architectures and distributed provisioning of services in mobile networks. His research interests also include distributed systems in the context to TCP/IP networks and peer-to-peer systems in particular. He holds a Dipl.-Ing. Univ. degree in Electrical Engineering from the Belgrade University (Serbia) and a PhD degree from the Swiss Federal Institute of Technology (École polytechnique fédérale de Lausanne, EPFL – Lausanne, Switzerland). The topic of his PhD thesis was "Trust and Reputation Management in P2P Systems". He has been involved in many IST projects such as Opelix, Evergrow, SmoothIT, Sail, Medieval and several projects funded by the Swiss and the German National Science Foundations.
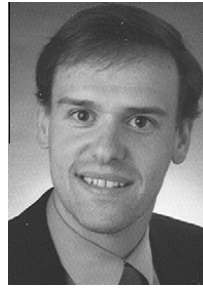
Wolfgang Kellerer is Director and head of the network research unit at NTT DOCOMO's European research labs in Munich, Germany. Before he joined DOCOMO in 2002, he has worked at the Institute of Communication Networks at Munich University of Technology (TUM). In 2001 he has been a visiting researcher at the Information Systems Laboratory of Stanford University, California. His current research interests are next generation mobile networks including optical communication, network virtualization, multimedia transport, advanced mobility solutions, overlay networks and service platforms.

Maximilian Michel got his Master degree in Computer Science at the Technical University of Munich (TUM) in 2006. After that he joined DOCOMO Communication Laboratories Europe in Munich. His research interests include networking, distributed systems, content distribution and peer-to-peer systems.