A Classification Scheme for the Peer Population of BitTorrent-like Peer-to-Peer Networks

Philipp M. Eittenberger, Udo R. Krieger
Faculty of Information Systems
and Applied Computer Science
Otto-Friedrich University,
Bamberg, Germany
Email: philipp.eittenberger@uni-bamberg.de

Seungbae Kim
School of Computer Science
and Engineering
Seoul National University,
Seoul, Korea
Email: sbkim@mmlab.snu.ac.kr

ABSTRACT

Mobile peer-to-peer (P2P) traffic is rapidly growing, but present P2P applications are not specifically designed to operate under mobile conditions. To assess the performance of the prevalent file sharing application BitTorrent in a mobile WiMAX network, we performed a measurement and analysis campaign. In this study, we use the obtained measurement traces to further investigate specific characteristics of this P2P network. In particular, we analyze the distribution of its peer population under mobile conditions and present a general classification scheme for peer populations in BitTorrent-like P2P networks.

I. INTRODUCTION

In recent years P2P networks evolved to the dominant source of traffic in the Internet [1]. Along with the evolution of a new generation of wireless networks, like WiMAX and 3GPP LTE, a shift to an increased user mobility can be witnessed. This development implies that users will have more and more opportunities to use all their accustomed applications wherever they are. That indicates that mobile P2P traffic will continue to grow rapidly in the next years (cf. [2]), but current P2P networks have been designed to operate in wired networks under stable conditions. Thereby, a growing need for new traffic models supporting P2P applications in a mobile environment and redesigned, mobility aware P2P protocols is emerging.

Kim et al. [3] conducted several measurements in Korea Telecom's WiMAX network in Seoul, Korea, to investigate the behavior of BitTorrent, one of the most popular P2P file sharing networks, under true mobile conditions. For this purpose, several measurements have been carried out while driving by bus and by subway through Seoul over multiple days. At each single measurement, a popular torrent with video content has been chosen for download and all the packet headers of the whole transmission have been captured. To allow comparison, the same files have been downloaded under static conditions in the WiMAX network and exemplary, with a host in a campus network connected to the Internet by an Ethernet link. For a detailed description of the measurement settings we refer to Section III-B.

In a WiMAX network disruptions occur during hand-overs and the wireless link conditions fluctuate due to signal fading. Hence, the main purpose of the measurement campaign was to investigate the impact of mobile conditions on the behavior of BitTorrent. In this paper, we further analyze the obtained measurement traces and present our insights into the BitTorrent peer population under mobile conditions.

The outline of the paper is as follows. We start with a discussion of related work in Section II. This is followed by an introduction to the BitTorrent network, its operational behavior and a presentation of our measurement methodology in Section III. Subsequently, we propose a classification scheme for peer populations in BitTorrent-like P2P systems and report our insights into the BitTorrent peer population in Section IV. Finally, we conclude our paper with the implications for the adaption of BitTorrent-like systems to wireless networks.

II. Related Work

Recently, several analytical models have been proposed to yield a deeper understanding of the P2P data dissemination among peers (e.g. [4] or [5]). Analytical models can provide precious insights, but are typically based on unrealistic assumptions, like global system knowledge. Therefore, we performed a measurement campaign, to reveal the actual system characteristics of the complex dissemination network of BitTorrent and to yield novel insights, which can be incorporated into new analytical models. Despite the fact that there can be found numerous measurement studies concerning the BitTorrent network (cf. [6] or [7] among many others), the first study, which analyzed the performance of such a file sharing network in a WiMAX environment under true mobile conditions, was performed by Kim et al. [3]. Sen and Wang [8] performed a large measurement campaign to analyze different P2P file sharing networks. In their paper they tried to characterize the peers of a particular mesh-pull P2P file sharing network. To examine their distribution, they plotted the traffic volume, duration of on-time and number of connections of the top 10 % of the investigated peer population on a log-log scale. From the plot they concluded that the distribution is heavy-tailed, but does not follow a power law. Subsequently, they concluded "that P2P traffic does not obey power laws". In this paper, we use the data traces obtained by Kim et al. to reveal deeper insights into the complex dissemination network of BitTorrent. We investigate the data access patterns between a single peer and the remote peer population, and show that certain parts of the peer population can be modeled by a power law distribution.

III. BitTorrent

As already mentioned, BitTorrent is one of the most popular P2P file sharing networks and itself a major source of the Internet traffic nowadays. The protocol specification is publicly available on the BitTorrent website [9]. Several distinct client applications, which implement the protocol, are available. Despite the fact that they differ in particular implementation details, they are able to exchange data with each other. BitTorrent is a representative of a mesh-pull P2P network. That means, it builds an overlay on top of the transport network based on a mesh topology. Additionally, the data dissemination is realized by a pull mechanism, i.e. on request. To enable fast data dissemination, BitTorrent uses the so called *swarming* technique, where each shared file is divided into smaller parts, called *chunks*, and then transmitted to (respectively received from) a multitude of peers (the *swarm*).

A. BitTorrent Operations in Detail

A *torrent* is the set of peers collaborating to share a single file. A peer can be in two different states, if the peer possesses already the complete file and uploads it to other peers, then it is called *seeder*. Otherwise, if it is still in the downloading phase, it is called *leecher*. To join a torrent, a peer needs some meta-information about it. This information is provided by a torrent file, containing all the information necessary to download the content, e.g. the number of chunks, hashes to verify their correctness, the IP of the tracker server etc. Typically this torrent file is retrieved from a website. Upon reception, the peer contacts the tracker server in the bootstrapping process, which provides an initial list of the latest remote peers. The peers of a torrent exchange messages to indicate the *chunks* that they already possess.

Two vital optimization problems have to be solved to achieve a high data throughput in a mesh-pull P2P network. At first, the choice which pieces should be requested from other peers, and subsequently, the selection which peers should be contacted for the data. BitTorrent addresses the first issue with a *rarest-first* algorithm, i.e. each peer maintains a list of pieces, that each of the remote peers has and builds an index of the pieces with the least number of copies. The *rarest pieces* are then requested from the remote peers. However, when a download is almost completed, the peer does not use the rarest-first algorithm; instead it sends requests for all of its missing pieces to all of its remote peers to increase the throughput. This is called *end-game mode*. The peer selection strategy is handled by the so called *choking algorithm*. To encourage peers to contribute their ressources for the data dissemination, a tit-for-tat mechanism is implemented to impede *free-*

riding, i.e. peers not contributing data to the network should not be able to achieve high download rates. Instead, this choking algorithm provides sharing incentives by rewarding peers who contribute data to the system. The algorithm determines the selection of peers to exchange data with. Peers that upload data at high rates are preferred. Once per *choking period*, usually every ten seconds, a peer evaluates the transmission rates from all the connected peers and selects a fixed number of the fastest ones, depending on its upload capacity. It will only upload data to these *unchoked* peers in this period. Data requests from other peers are denied in this period, i.e. those peers are *choked*. Another important part of the algorithm is the *optimistic unchoking* behavior: every 30 seconds one peer is randomly chosen and will be unchoked. This is meant to explore new peers with even higher upload capacities and as a side effect ensures data dissemination to low-capacity peers. Once a leecher has finished the download and enters the *seeding* state, it follows a different unchoke strategy. In most of the implementations peers in seed state unchoke peers with the highest download capacity to optimize the general dissemination performance of the network and to maintain high upload utilization.

B. Measurement of BitTorrent in a WiMax Network

The analyzed measurement traces have been captured in March, 2010 in Seoul, Korea and were firstly presented in [3]. The measurements have been carried out on four different days in parallel, i.e. on each day all the measurement runs started at the same time, in four different scenarios, three WiMax settings and one in a Ethernet environment. The throughput of the WiMax network ranges from 30 to 50 Mbps, and a base station typically covers a radius between 1 and 5 km. Three laptops equiped with WiMax USB dongles where used for the WiMax measurements and one desktop computer for the reference Ethernet measurement. *Vuze* [10] was used as the BitTorrent client in all measurement runs. For the measurement some popular sitcoms served as torrents, which had at least 300 seeds, with a file size ranging from 300 to 400 MB. An important fact to note is the over-provisioning with seeding capacity in the measurement runs to ensure a high data throughput. Of course, on each day the same torrent was chosen to download in all four different settings. One WiMax measurement was conducted stationary, i.e. the measurement peer was located **static**ally about 800 meters away from its base station. Therefore, the signal strength was stable, but not strong. For the next scenario one peer traveled about 12 km in a **subway** train through Seoul while conducting the measurement. The duration was about 20 minutes. In the last WiMax scenario the peer took a **bus** ride through Seoul, which lasted about 30 minutes and the distance of the route was about 11 km. In both mobile scenarios the link quality fluctuated highly and it even occurred that the WiMax connection was completely lost due to hand-overs in between the base stations, and thereby, the peer got a new IP address in some of the measurement runs. An **Ethernet** measurement in a 100 Mbps LAN was conducted on the university campus as a reference to allow comparison. For a more detailed description, we refer to Kim et al.'s study [3]. The main results of this study will be also presented shortly. The WiMax peers suffer from poor connectivity, the connections to the peers are less stable and the connection duration is shorter as opposed to the reference Ethernet measurement. In all WiMax settings the duration of the file download took 4 to 5 times longer than in the Ethernet measurement. The signaling traffic has been increased in all WiMax measurements by a factor of 100 %. The mean of the average download rates in the WiMax measurements ranges from roughly 240 to 400 Kbps, as opposed to a mean of 1930 Kbps in the Ethernet runs. One reason for the poor performance of BitTorrent in the WiMax network is certainly the fluctuating signal strength, but the hand-overs in the mobile measurements have a negative impact on the performance of TCP transmissions, too.

IV. Classification of the Peer Population

A peer in a mesh-pull P2P network is typically connected to more than a thousand different peers in just tens of minutes. Hence, it is vital for the understanding of the inherent hierarchical structure of a mesh-pull topology to explore the preference relationship among the peer population. We have shown in our previous work (cf. [11]) that it is possible to classify the peer population of P2P streaming networks. In this study we investigate

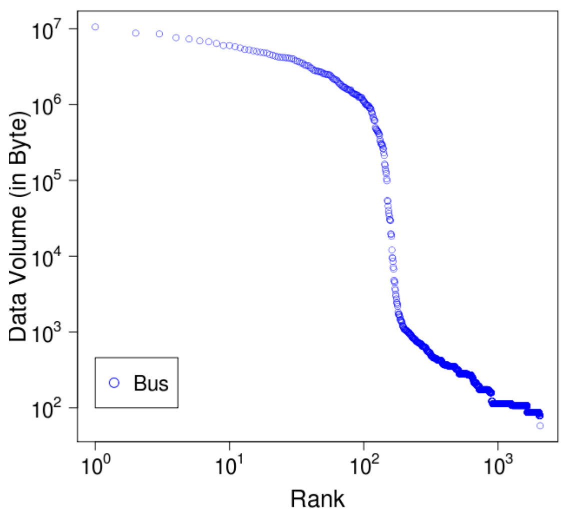

Figure 1: Inbound data distribution
(bus trace of March, 17).

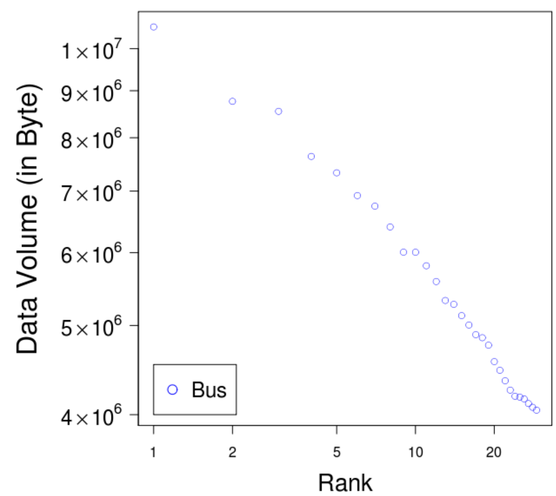

Figure 2: Inbound data distribution in the profitable region (bus trace of March, 17).

the aggregated conversation model of superimposed flows in inbound and outbound direction to a home peer, i.e. the traffic volume generated by the superimposed flows from and to the distinct feeder peers p_i of the dissemination flow graph G_V. To clarify our terminology, we regard a *conversation* as bidirectional data exchange between two endpoints, in this case peers. A *flow* represents the directed data transfer, which can be identified by the traffic direction, the IPs of both peers, the port numbers of the used transport protocol and a given timeout value to differentiate widely disparate flows. A *stream* represents the aggregated set of flows sent from one peer to another.

Using the captured data traces, it allows us to describe the exchange of chunk sequences among the peers by appropriate teletraffic models. We use the amount of exchanged traffic, i.e. received and sent bytes, in all streams as a metric for the further classification of the peer population. Since we are interested in the contribution of each peer and the variance among different peers' contribution, it is a convenient choice to rank all peers according to their contribution and then identify those with high contributions. Therefore, we sort each stream by the number of transferred bytes in descending order. If we arrange the streams $\phi(p_i, p_j)$ according to their number of exchanged bytes on a log-log scale, where p_j is representing the home peer, i.e. the measurement host, and $p_i, i \in \{1, ..., n\}$, denotes the feeding peer population, we can realize a hierarchy of the peer population. To clarify our concept, we use the WiMax bus trace of March, 17 as an example. By using the rank ordering technique, we plotted the distribution of the incoming streams in Fig. 1. Several distinct regions can be spotted in this distribution. The *profitable region* consists of the *top peers* and it's body resembles in all captured WiMAX traces asymptotically a straight line. In this example, the region ranges roughly from 4,000,000 bytes to 10,000,000 bytes. The upper limit of this region is given by the file size or for streaming P2P networks by the session length. Top peers contribute the largest share of the total data volume, i.e. most of the data is received in the conversations with these peers. In the exemplary trace this region consists of 29 peers, which sent 47.80 % of the total data volume.

The next region, ranging from 20,000 bytes to 4,000,000 bytes, is called the *productive region*, because it is likely that the streams do not only consist of signaling overhead, but also of useful data. This means, that chunks have been transferred, but the ratio of signaling overhead is worse than

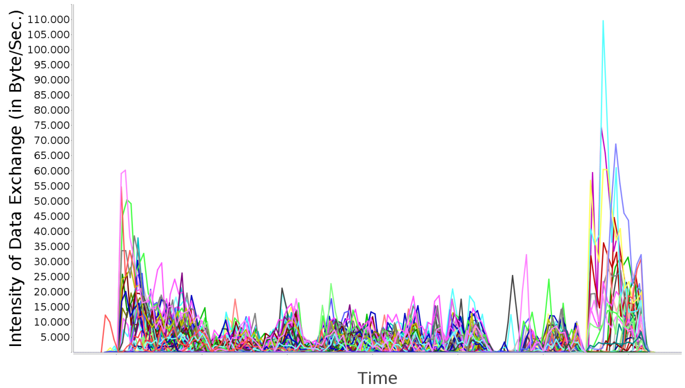

Figure 3: Intensity of the inbound traffic (Bus trace of March, 17).

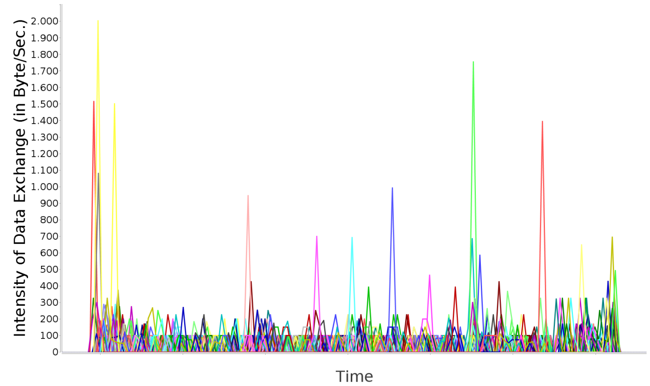

Figure 4: Intensity of the inbound traffic in the *unproductive region* (Bus trace of March, 17).

before. This region is built by *ordinary peers*. In this trace the 129 ordinary peers sent 52.03 % of the total data volume. Thus, 99.84 % of the total data volume has been transferred by the 158 peers of the profitable and productive region. All other streams below this region can be regarded as almost useless for the operations of the P2P network due to their minimal contribution towards the volume of useful data. They consist mainly of signaling overhead, e.g. connection establishment and maintenance. Hence, we call the next region *unproductive*, which is inhabited by *futile peers*. The horizontal lines at the end of this region mainly consist of unsuccessful connection attempts. Out of the 2060 incoming streams 1902 streams lie in the unproductive region. To investigate this inefficiency more thoroughly, we visualize the intensity of the incoming streams, i.e. the amount of bytes per 10 second intervals, with the help of traffic analyzer Atheris [12] (see Fig. 3). The effects of the hand-overs are clearly visible, precisely when the intensity of the incoming streams tends towards zero. Also, due to the end-game mode, at the end of the trace, the intensity increases dramatically. When we visualize only the streams of the unproductive region, see Fig. 4, one realizes that the incoming streams of the unproductive region continue over the whole trace. This is not a big problem, when BitTorrent is operating in a wired environment, like Ethernet, but in a WiMax scenario this can be a high burden for wireless access points, especially when multiple clients use simultaneously a BitTorrent-like application. For the adaption of the BitTorrent protocol to a wireless environment, we recommend to intensify the data exchange with peers in the profitable region and restrict it to peers in the productive region, and thereby, avoiding the many useless connections in the unproductive region.

So far, we have explained our observations by one representative trace, but to allow a comparison, we have additionally plotted the distributions of the inbound and outbound streams in all the gathered traces, see Fig. 5 and 6. Apart from the Ethernet measurement on March 16, the bodies of all the data access distributions have the same shape on the log-log scale. We see very clearly the influence of the *choking* algorithm on the head of the distributions in all Ethernet measurements, but also in the outbound traffic distribution of the WiMax traces. Very few peers, between three and six, receive by far the biggest share of the data. Additionally, in the Ethernet scenario, only a few peers sent most of the data to the measurement host. However, this pattern changes in all WiMax traces and the head of the inbound data distribution becomes flat. This implies that the received data volume is more evenly distributed over the peer population. The cause for this change is due to the limited upload capacity of the WiMax peers. They are choked more often and rely mainly on the optimistic unchoke behavior, in order to successfully complete their download. Thereby, the download performance suffers in all WiMax scenarios.

We interpret the number of transferred bytes of a stream $\phi(p_i, p_j)$ as realization x_i of an equivalent

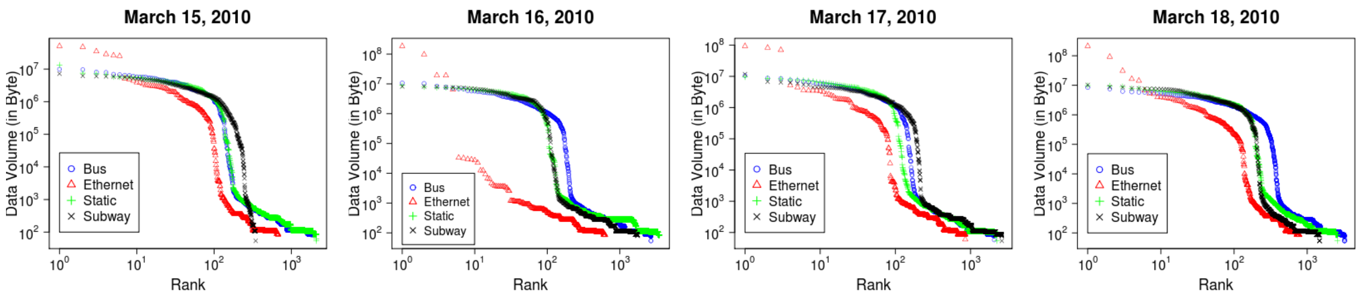

Figure 5: Inbound data access distribution.

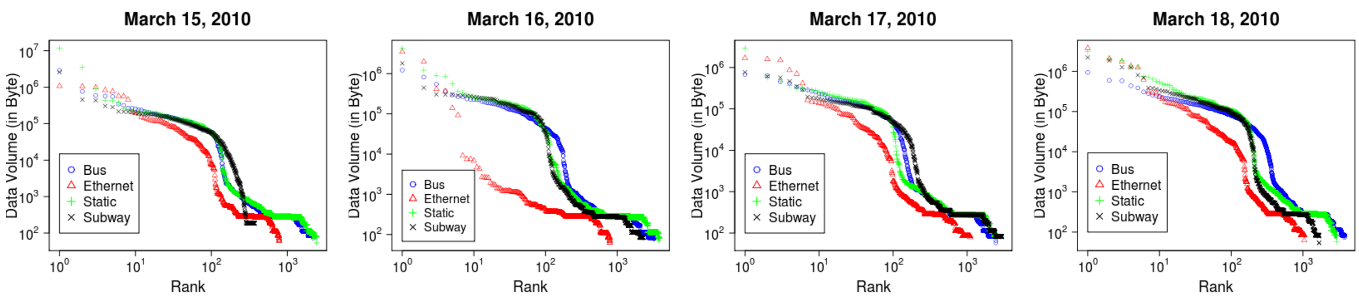

Figure 6: Outbound data access distribution.

income $X_i \in \mathbb{R}$ of the home peer p_j. Considering the overhead for establishing and maintaining the connection to the feeded peer p_j as costs, only the connections with top peers are really profitable. Thus, the main focus of interest is given by the distribution of the top peers. Since the feeding peer population of this region contributes the largest proportion (approx. 50 % in the exemplary bus trace of March, 17) of useful data with the best signaling overhead ratio. The asymptotic straight line of the *profitable region* on the log-log scale (see Figure 2) indicates that the head of the distribution follows a power law. Thereby, we can use a generalized Pareto distribution to model this region of the peer population with a random variable X and its sample $\{x_1, ..., x_n\}$. We denote the distribution function of this generalized Pareto model by

$$F(x) = 1 - (1 + k\frac{x-\mu}{\sigma})^{-1/k} \quad for \quad k \neq 0,$$

with $\mu \leq x \leq \mu - \sigma/k$ for $k < 0$. To investigate the distribution of the top peers, we set the minimum x_{min} to the lower bound of the profitable region, i.e. $x_{min} = 3,981,064$ bytes. x_{min} is obtained with Clauset's estimator (cf. [13]), which chooses the value of $\hat{x}_{min}$ such that the probability distribution of the measured data and the best-fit power law model is as similar as possible above $\hat{x}_{min}$. Hereby, we consider only the flows from the top peers $\phi(p_i, p_j)$, with $p_i, i \in \{1, ..., n\}$ and $n = 29$, feeding the home peer p_j. Using the transferred amount of bytes $x_1 \leq x_2 ... \leq x_n$, we can determine the scaling parameter $\hat{\alpha} = 4.281707$ by Newman's estimate [14]

$$\hat{\alpha} \simeq 1 + n \left[\sum_{i=1}^{n} ln \frac{x_i}{x_{min} - \frac{1}{2}} \right]^{-1}.$$

It is obvious that the Pareto model can only be applied for the profitable region, since the distribution of the peers in the productive region is not a straight line on the log-log plot, but has a flat head and a steep tail. Such a rank distribution indicates a Weibull distribution. The cumulative distribution function (cdf) for the Weibull distribution is given by

$$F(x) = 1 - e^{-(x\alpha)^k} \quad for \quad x \geq 0,$$

where α is the scale parameter and k the shape parameter. Both parameters are constant and in the

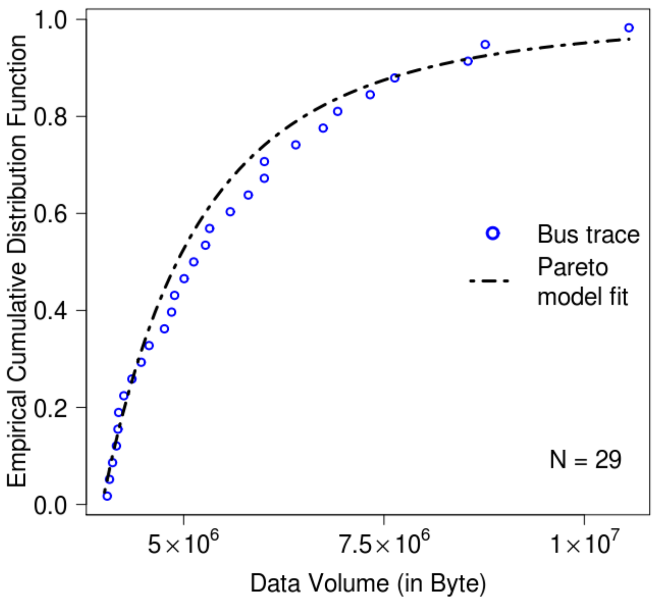

Figure 7: Cdf of the inbound data distribution in the *profitable region*.

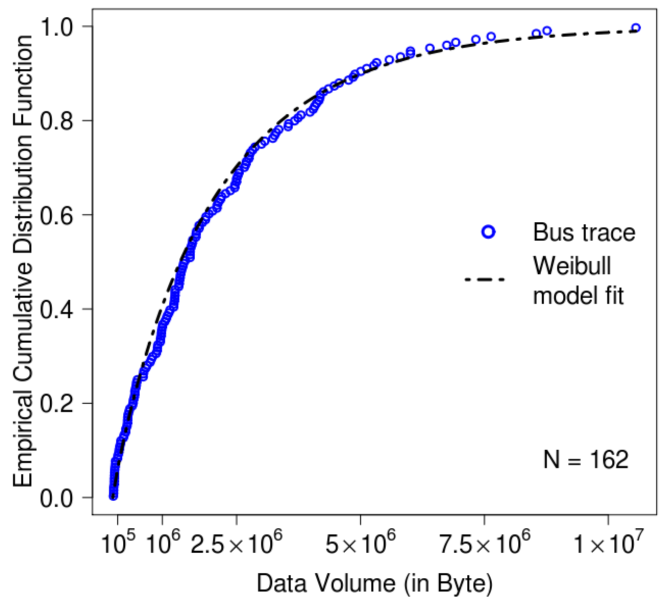

Figure 8: Pdf of the inbound data distribution in the *productive region*.

analyzed trace $\alpha = 1.015688$ and $k = 2259901$. To get the parameters, we used the maximum likelihood estimation method. Since the scale parameter α is close to 1, the increase of the data volume per peer over time is fairly constant. Fig. 7 and 8 show the cumulative distribution function (cdf) of the received data volumes in both regions and the dotted lines show the fitted models, i.e. the Pareto, respectively the Weibull distribution.

As a goodness-of-fit metric we use the Kolmogorov-Smirnov test, which is the largest vertical distance between the fitted and actual cumulative distribution functions, measured in percentiles. We obtain a P-value of 0.9134 for the Pareto model fit of the profitable region and a P-value of 0.9563 for the fitting of the productive region to the Weibull distribution and with a significance level of $\alpha = 0.01$ the null hypothesis can in both cases not be rejected. This constitutes a strong indication that the observed sample data obey a generalized Pareto respectively a Weibull distribution. Regarding the distribution of the peers in the profitable and productive region, the same observations can be made in all the other captured WiMAX traces.

V. Conclusion

To the best of our knowledge, this is the first study, which investigates the data access patterns of BitTorrent and tries to fit the distribution of the feeding peer population to a model. In our work, we have additionally introduced a novel scheme to classify the peer population of BitTorrent-like P2P networks into different categories. We have shown that there are strong indications that a particular part of the peer population obeys a power law and that its distribution can be modeled by a generalized Pareto model. Furthermore, when we investigated the data distribution of the productive region, we found strong indications that the part of the peer population, which contributes nearly the complete data volume, can be modeled by a Weibull distribution. The limitation of this study is the single point of view regarding the measurements, but in future work we plan to support our results with a distributed measurement campaign. We have seen that the BitTorrent protocol and especially the choking algorithm is not well adapted to the fluctuating conditions in a WiMax environment. Therefore, we propose following recommendations for the adaption of BitTorrent-like systems to wireless networks. A very simple part of the solution could be using another client instead of Vuze. Iliofotou et al. [15] have shown that μTorrent achieves on average a 16 % better download performance compared to Vuze. One reason for the better performance might be the limitation of the amount of open connections to peers by μTorrent. In general, this might also be a good recommendation in wireless scenarios, and thereby, not to overload the base stations with too many open connections. Another proposition would be to use UDP instead of TCP as a transport protocol, since especially TCP, as a connection oriented protocol, suffers from the fluctuating link conditions and by the hand-overs in a mobile wireless scenario. Finally, Lehrieder et al. [16] investigated the positive effect of caches in a BitTorrent network. As recommendation for network operators, supplying a dedicated infrastructure with local caches to support the data dissemination of BitTorrent-like networks can dramatically increase the data throughput, but at the same time reduce the load on the own infrastructure by reducing inter-domain traffic. This proposal is very important with regard to the limited upload capacity in a WiMax network, since the choking algorithm is not well adapted to conditions of wireless networks and the peers rely heavily on the optimistic unchoking behavior to complete their download. Therefore, supplying a dedicated infrastructure with local caches could foster the dissemination performance of BitTorrent-like systems in wireless networks.

REFERENCES

[1] Cisco Systems, "Cisco visual networking index: Forecast and methodology, 2009-2014," *White Paper*, 2010.

[2] ——, "Cisco visual networking index: Global mobile data traffic forecast update, 2009-2014," *White Paper*, 2010.

[3] S. Kim, X. Wang, H. Kim, T. T. Kwon, and Y. Choi, "Measurement and analysis of BitTorrent traffic in mobile WiMAX networks," in *10th IEEE International Conference on Peer-to-Peer Computing 2010 - IEEE P2P 2010*. IEEE, 2010, pp. 49–52.

[4] D. Qiu and R. Srikant, "Modeling and performance analysis of bittorrent-like peer-to-peer networks," in *Proceedings of the 2004 conference on Applications, technologies, architectures, and protocols for computer communications (SIGCOMM '04)*. ACM, 2004, pp. 367–378.

[5] G. Dán and N. Carlsson, "Dynamic Swarm Management for Improved BitTorrent Performance," in *Proc. International Workshop on Peer-to-Peer Systems (IPTPS '09)*, Boston, MA, 2009.

[6] L. Guo, S. Chen, Z. Xiao, E. Tan, X. Ding, and X. Zhang, "Measurements, analysis, and modeling of BitTorrent-like systems," in *IMC '05: Proceedings of the 5th ACM SIGCOMM conference on Internet Measurement*. Berkeley, CA, USA: USENIX Association, 2005, pp. 35–48.

[7] A. R. Bharambe, C. Herley, and V. N. Padmanabhan, "Analyzing and improving a BitTorrent networks performance mechanisms," in *25th IEEE International Conference on Computer Communications (INFOCOM 2006)*, 2006, pp. 1–12.

[8] S. Sen and J. Wang, "Analyzing peer-to-peer traffic across large networks," *IEEE/ACM Transactions on Networking*, vol. 12, no. 2, pp. 219–232, 2004.

[9] Bittorrent. [Online]. Available: http://www.bittorrent.org/beps/bep_0003.html

[10] Vuze. [Online]. Available: http://www.vuze.com/

[11] N. M. Markovich, A. Biernacki, P. M. Eittenberger, and U. R. Krieger, "Integrated measurement and analysis of peer-to-peer traffic," in *Wired/Wireless Internet Communications, 8th International Conference*. Springer, 2010, pp. 302–314.

[12] P. M. Eittenberger and U. Krieger, "Atheris: A first step towards a unified peer-to-peer traffic measurement framework," in *19th Euromicro International Conference on Parallel, Distributed and Network-Based Computing (PDP 2011)*. Euromicro, 2011.

[13] A. Clauset, C. R. Shalizi, and M. E. J. Newman, "Power-law distributions in empirical data," *SIAM Reviews*, 2007.

[14] M. E. J. Newman, "Power laws, pareto distributions and zipf's law," *Contemporary Physics*, vol. 46, pp. 323–351, 2005.

[15] M. Iliofotou, G. Siganos, X. Yang, and P. Rodriguez, "Comparing bittorrent clients in the wild: the case of download speed," in *Proceedings of the 9th international conference on Peer-to-peer systems(IPTPS'10)*. USENIX Association, 2010.

[16] F. Lehrieder, G. Dán, T. Hoßfeld, S. Oechsner, and V. Singeorzan, "The Impact of Caching on BitTorrent-like Peer-to-peer Systems," in *10th IEEE International Conference on Peer-to-Peer Computing 2010 - IEEE P2P 2010*, Delft, Netherlands, 2010, pp. 69–78.