

# On the Double-Faced Nature of P2P Traffic

Raffaele Bolla, Marco Canini, Riccardo Rapuzzi, Michele Sciuto

*DIST - Department of Communication, Computer and System Sciences, University of Genoa*

*Via Opera Pia 13, 16145, Genova, Italy*

*{raffaele.bolla, marco.canini, riccardo.rapuzzi, michele.sciuto}@unige.it*

## Abstract

*Over the last few years, peer-to-peer (P2P) file sharing applications have evolved to become a major traffic source in the Internet. The ability to quantify their impact on the network, as a consequence of both signaling and download traffic, is fundamental to a number of network operations, including traffic engineering, capacity planning, quality of service, forecasting for long-term provisioning, etc.*

*We present here a measurement study on the characteristics of the traffic associated with different P2P applications. Our aim is to offer useful insight into the nature of P2P traffic, which we consider a step toward building P2P traffic aggregates generators in simulative environments. We show that P2P traffic can be divided into two distinguished behavioral profiles, which, independently of the application protocol, present significant differences in the average and standard deviation of four measurements: arrival times, durations, volumes and average packet sizes of P2P conversations. These profiles well represent the typical behavior of signaling and download traffic. Based on our findings, we argue that, if such distinction is not taken into account, the statistical measurements needed to model P2P traffic aggregates would result biased, and potentially bring to misleading results.*

## 1 Introduction

Over the last few years, peer-to-peer (P2P) file sharing applications have evolved to become a major traffic source in the Internet. Plus, the successful P2P paradigm is now used by an increasing number of systems to offer efficient and scalable services ranging from content distribution, network TV, voice-over-IP to software updates. An important consequence related to the appearance of P2P systems is related to the high traffic volumes caused by these applications. The ability to quantify their impact on the network, as a consequence of both signaling and download traffic, is fundamental to a number of network operations, including

traffic engineering, capacity planning, quality of service, forecasting for long-term provisioning, etc.

We present here a measurement study on the characteristics of the traffic associated with different P2P applications. We identify P2P traffic in our data sets by using a payload classifier based on pattern matching. We develop a characterization of this traffic that focuses towards its behavior as it is experienced at the network layer. In this work, which grounds up on our previous work in [1], we define a measurement, Content Transfer Index, to distinguish two classes of behavior for P2P traffic: the download and the signaling traffic profile. Finally, we show that these two profiles present relevant differences in the average and standard deviation of four measurements: arrival times, durations, volumes and average packet sizes. Based on our findings, we argue that the statistical measurements needed to model P2P traffic aggregates would result biased, and potentially bring to misleading results, if its two-faced nature is not taken into account. We believe this work gives a sincere contribution towards building P2P traffic aggregates generators in simulative environments.

Our work is similar to [10], but it is different in that: (i) we don't limit our study to the eDonkey protocol, and (ii) we identify the P2P traffic in our data sets by using a payload classifier, rather than the application port numbers. As their results are comparable with the ones reported in this paper, we consider promising our approach, which doesn't rely on the protocol opcode information to distinguish download from signaling traffic. In another related work [2], the authors have analyzed BitTorrent traffic characteristics, using an instrumented version of the official client. They suggest that conversation durations and sizes can be reasonably well approximated by the lognormal distribution, which is also confirmed in our results. A complementary work is [3], where the authors model and characterize BitTorrent-like systems by using measurements at the application layer.

The remainder of this paper is organized as follows. In Section 2, we describe how we identified P2P traffic in our data sets. In Section 3, we present our characterization of

P2P traffic. The results of this characterization applied to a number of measurements are shown in Section 4. Finally, Section 5 concludes the paper.

## 2 Data collection

We analyzed traces contemporarily collected for 46 consecutive hours, at two different links of the university’s campus network. For both traces, we captured every packet seen on each direction of the links along with its full payload and we removed the link layer header (i.e., Ethernet). To capture the first trace (DEPT), the monitor was located on the link connecting our department to the campus network. The second trace (GENUA) was captured by monitoring the campus connection to the Internet. We ignore the TCP connections for which we don’t observe the triple handshake.

Table 1 lists general workload dimensions of our data sets: counts of distinct source and destination IP addresses, the number of flows, packets, and bytes observed. In this study we define flows as unidirectional, while we use *conversation* to denote bidirectional traffic, i.e., a conversation is composed by two flows: traffic from A to B, and traffic from B to A. Each flow is always identified by two end points consisting of {IP, port} pairs and the transport level protocol. In the case of the UDP, we set a timeout of 60 seconds to determine the end of a conversation.

We used an open source tool, namely *ipp2p* [4], to classify the P2P traffic in our traces. It identifies P2P flows via pattern matching, i.e., searching the payload content of the packets for known protocol signatures. This classifier acts every time a packet is received, and marks a conversation as identified as soon as a known pattern is found in at least one of its directions. For scalability reasons, only up to the first  $N$  packets of each conversation are tested.  $N$  is a user configurable parameter, that we set to 10 as done in [9]. Because this tool is not available as an off-line, trace processing tool (it is originally meant to be deployed in the Linux’s iptables firewall for traffic shaping purposes), we ported its source code to the Click Modular Router [7], which we used as a viable analysis framework [6]. We validated our version of the tool against its original version by comparing the results obtained from the classification of our data sets. The outputs were indeed the same. Finally, we collated the classification results obtained with our *ipp2p* classifier with the output of the payload classifier used to validate BLINC [5], obtaining very close results on our data sets.

In this paper, we present results obtained using the GENUA data set; the results for the DEPT data set are similar and not presented here due to lack of space.

Table 2 presents the volumes of P2P traffic in GENUA trace, divided by P2P application. The values express the percentage with respect to the total amount of P2P traffic. Both the traces are 46 hours long.

Set	Src. IP	Dst. IP	Flows	Packets	Bytes
DEPT	376.7 K	328.3 K	16.8 M	171.3 M	103.3 G
GENUA	4.2 M	2.5 M	102.0 M	1.4 G	883.0 G

**Table 1. General workload dimensions of our traces**

Protocol	Flows [%]	Packets [%]	Bytes [%]	Payload Bytes [%]
eDonkey	83,78	46,19	42,12	41,88
BitTorrent	13,65	37,19	45,72	46,33
DC	0,34	7,50	9,99	10,23
Gnutella	2,21	9,11	2,16	1,56

**Table 2. Breakdown by protocol of P2P traffic in the GENUA data set**

In the remainder of the paper we mainly focus on the traffic generated by eDonkey and BitTorrent, since this constitutes the majority in our campus network<sup>1</sup>. However, we also report results about the traffic associated with Gnutella and DirectConnect.

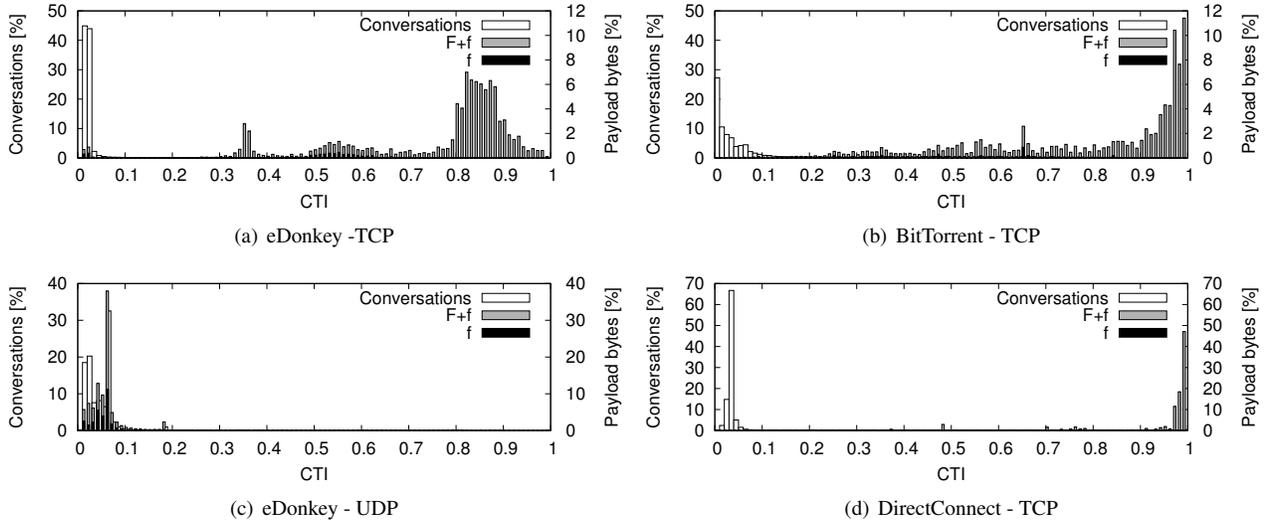
## 3 P2P traffic characterization

P2P traffic can be roughly divided into download traffic and signaling traffic: the first is caused by the transfer of content, the latter is mainly due to the presence of an overlay network, and possibly a search service.

A way to accurately differentiate download from signaling traffic would be to implement a protocol analyzer. Although one can leverage existing tools, e.g. *binpac* [8], to build protocol analyzers, this solution has several drawbacks: (i) it requires specific knowledge of P2P protocols; (ii) it demands access to the payload of each packet; (iii) building analyzers for a significant number of protocols is definitely time consuming. Moreover, it is our desire to offer a characterization of P2P traffic that focuses towards its behavior as it is observable at the network level through simple passive measurements, including volumes of transferred content, conversations’ arrival times and durations, and average packet sizes.

In our approach we identify two behavioral classes that we shall refer to as the download and the signaling traffic profile. To some extents we’re abusing the terminologies, as it is possible, though not common, that a download conversation can exhibit the typical characteristics of signaling

<sup>1</sup>P2P traffic is believed to be hazardous for networks, and our campus network makes no exception. We are aware that a filtering system has been deployed, realizing traffic shaping for the most common P2P applications.



**Figure 1. Relationships between the conversations, payload bytes and the CTI. The histograms are plotted with CTI bin size 0.01**

traffic<sup>2</sup> and vice versa. However, we are mostly interested in distinguishing the conversations by considering the actual effects that they pose on the network, treating P2P with generality, rather than by using an accurate identification of their role.

We ground our approach on a simple intuition. The idea is that, during a file download, a peer, on average, gets packets filled up to the maximum segment size (MSS) and sends back fewer packets to acknowledge the received data. Even when the peers are exchanging pieces of a single file with one another (as realized in BitTorrent), causing balanced conversations, still, the average payload size tends to reach the MSS. Conversely, the signaling conversations are characterized by a flatter profile, consisting of a more even count of exchanged bytes and packets. The accurate identification of eDonkey conversations<sup>3</sup> in our traces supports our intuition.

In order to quantify the differences between these traffic profiles we formalize a measurement that offers a statistical characterization of P2P traffic. We define the Content Transfer Index (*CTI*) of a conversation  $C$  as:

$$\frac{F}{f+F} \cdot \frac{\bar{P}}{\text{MSS}(C)} + \frac{f}{f+F} \cdot \frac{\bar{p}}{\text{MSS}(C)} \in [0, 1],$$

where  $F, f$  are the lengths of the two flows constituting  $C$ , such that  $f \leq F$ . One can use three different flow fea-

<sup>2</sup>E.g., early truncated downloads are not clearly distinguishable from signaling conversations.

<sup>3</sup>As done in [10], we classify eDonkey conversations to be download if one of its packets contains at least one of the eDonkey protocol opcodes ‘OP\_SENDINGPART’ or ‘OP\_COMPRESSEDPART’.

tures to represent its length: the packet count, the count of payload bytes and the count of headers and payload bytes. We present here the results obtained by using the count of payload bytes as the flow length.  $\bar{P}$  and  $\bar{p}$  represent the average number of payload bytes per packet calculated for the flow with length  $F$  and  $f$  respectively. The MSS of the conversation  $C$  is expressed as  $\text{MSS}(C)$ . For the UDP, we assume that the MSS corresponds to the maximum transfer unit (MTU) minus the IP and UDP headers’ lengths.

Hence, given a pair of end points  $\{A, B\}$ , the CTI gives an indication of how the content is transferred. At the opposite end of the spectrum there are two distinct traffic profiles:

- when the conversation is flatter or balanced (i.e., A and B exchange an even quantity of content, mainly using packets whose payload size is far from the MSS), the CTI’s value tends to zero;
- when the conversation is rich of content which is either transferred from a single end point that dominates the conversation (unbalanced), or efficiently exchanged between the two end points, the CTI’s value tends to one.

Figure 1(a), 1(b), 1(c) and 1(d) present a graphical visualization of the relationships between the conversations, payload bytes and the CTI of eDonkey TCP, BitTorrent TCP, eDonkey UDP and DirectConnect TCP conversations respectively. Each figure shows three overlapping histograms, symbolizing the following figures corresponding to the same CTI range: the number of conversations, the

summation of the minimum and maximum length flows of the conversations (denoted with  $F + f$ ) and the summation of just the minimum length flows (denoted with  $f$ ). The number of conversations is normalized to the total number of conversations, while both summations of minimum and maximum length flows, and of just the minimum length flows are normalized to the total count of payload bytes. All the graphs but Figure 1(c) clearly show two distinctive profiles: containing most of the conversations, the signaling profile is having CTI values below 0.2, whereas above 0.2, the download profile is dominated by the payload bytes. Also note in the signaling profiles that the conversations are quite balanced.

As expected, the eDonkey UDP conversations, that are carrying extended searches, generate a graph in which no conversation has CTI value above 0.2. The same happens in the graph of Gnutella TCP conversation (that we omit for the lack of space), as the download traffic of this protocol is efficiently filtered by the campus border router. Figure 1(d) contains a drastic example of how the two traffic behave differently: almost all the traffic identified as DirectConnect, accounting for 10.23% of the total P2P traffic, is moved by just very few conversations (0.02% of the total P2P conversations), while the 94% of DirectConnect conversations transfer an extremely small amount of data.

Throughout the rest of the paper, we divide P2P traffic into download vs. signaling by using the CTI's value. In particular we use a threshold to distinguish the two classes: a conversation with a CTI's value above the threshold is marked as a download conversation, while a value below the threshold determines a signaling conversation. We set the threshold to 0.2. This value was already used in our previous work in [1] which offers a validation of the CTI, and it is visually derivable from Figure 1.

## 4 Results

In this section we present the results obtained by measuring arrival times, durations, volumes and average payload sizes of the download and signaling P2P conversations. For each measure, we report the measured data along with the best fitting<sup>4</sup> statistical distribution and its parameters. We considered the following candidate distributions: exponential, Pareto, lognormal, and Weibull. Note that we present the parameters in order to offer a numerical comparison of the empirical data across different protocols and traffic profiles. Because these are preliminary results of this work, we are not suggesting to use them directly in the modeling of P2P traffic aggregate generators, as this still require a deeper investigation.

**Arrival times.** Figure 2 shows the CCDFs of the down-

<sup>4</sup>By using the nonlinear least-squares method.

load and signaling conversation arrival times, for eDonkey and BitTorrent TCP traffic. Table 3 lists their average and standard deviation. In all the four graphs, the best fitting distribution is Weibull. Its parameters are more comparable when looking at the same traffic profile rather than the same protocol. It is clear from Table 3 that measuring the whole P2P traffic for a given protocol leads to inaccurate values of the average and standard deviation of conversation arrival times, as there is a significant difference between the values measured for download and signaling traffic.

Conversation type	Avg. [s]	Std. dev. [s]
eDonkey - download	1.76	2.34
eDonkey - signaling	0.11	0.16
eDonkey - all	0.10	0.15
BitTorrent - download	3.00	4.88
BitTorrent - signaling	1.01	1.51
BitTorrent - all	0.75	1.13

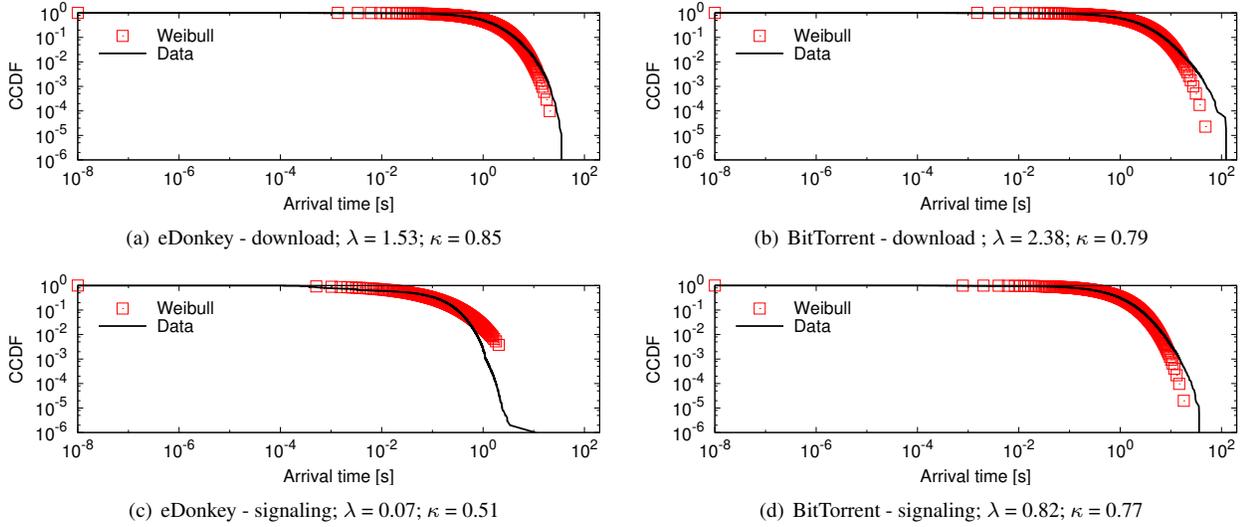
**Table 3. Average and standard deviation of the conversation arrival times [s]**

**Durations.** The CCDFs of the download and signaling conversation durations are shown in Figure 3, plotted for eDonkey and BitTorrent TCP traffic. The lognormal is the best fitting distribution in all the four graphs. Table 4 lists their average and standard deviation. Interestingly the profiles of the download conversation durations are quite similar and the parameters of the fitted lognormals are numerically very close, while the behavior of signaling traffic is not so similar. We believe this is mostly due to the absence of a search service in the BitTorrent system, which is on the contrary available in the eDonkey network. Once again, both the graphs and Table 4 indicate that the behavior of download and signaling traffic is different. Measuring the P2P traffic without distinguishing download from signaling traffic leads to inaccurate values for the average and standard deviation of conversation durations.

Conversation type	Avg. [s]	Std. dev. [s]
eDonkey - download	276.76	784.79
eDonkey - signaling	51.64	490.36
eDonkey - all	64.56	514.53
BitTorrent - download	749.43	2076.03
BitTorrent - signaling	141.10	1513.28
BitTorrent - all	293.85	1693.15

**Table 4. Average and standard deviation of conversation durations [s]**

**Data Volumes.** Figure 4 reports the CCDFs of the conver-



**Figure 2. CCDF of the conversation arrival times.  $\kappa$  and  $\lambda$  are the Weibull parameters**

sation volumes, plotted for eDonkey and BitTorrent download and signaling traffic. All the profiles are best fitted by the lognormal distribution. The differences in the parameters well reveal what is already clear in the graphs: download and signaling traffic exhibit different profiles. The download conversation volumes are very similar for eDonkey and BitTorrent, while the signaling ones are, on average, one order of magnitude smaller than the download conversation volumes for both protocols. Also note that we identified some download conversations that are transferring hundreds of megabytes of data.

**Average payload sizes.** Table 5 presents the average and standard deviation of the download and signaling conversation average payload sizes, plotted for eDonkey and BitTorrent TCP traffic. The average payload size of a single conversation has been calculated as the mean of the average payload sizes of both flows for a signaling conversation, or as the maximum length flow’s average payload size in the case of a download conversation.

Figure 5 present the relationships between the conversations, volumes and the average payload size for the signaling conversations of eDonkey and BitTorrent. The graphs, which are realized using the same structure of Figure 1, show that there is significant difference in terms of the average payload sizes used in these protocols. In particular, note that the conversations are more balanced for eDonkey. We believe this reflects the architectural differences of the two P2P networks.

However, for both protocols, there is no signaling conversation having average payload size greater than 300 bytes.

Conversation type	Avg. [bytes]	Std. dev. [bytes]
eDonkey - download	894.611	290.37
eDonkey - signaling	30.70	12.63
eDonkey All	55.05	105.22
BitTorrent - download	792.89	257.57
BitTorrent - signaling	41.08	45.51
BitTorrent All	138.83	185.24

**Table 5. Average and standard deviation of the conversation average payload sizes**

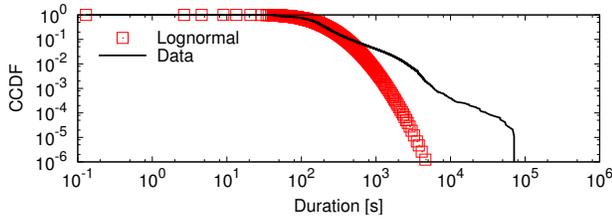
## 5 Conclusion

In this paper, we have presented a measurement study of P2P traffic. We have first applied the Content Transfer Index to distinguish the traffic into two classes of behavior: the download and the signaling traffic profile. Based on this distinction, we have realized a number of measurements and shown that there are significant differences in their average and standard deviation from the case where the distinction is not taken into account. In particular, our measures result unbiased.

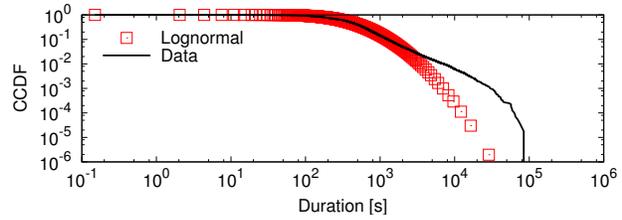
These results indicate that the double-faced nature of P2P traffic should be taken into account in modeling P2P traffic aggregates.

## Acknowledgment

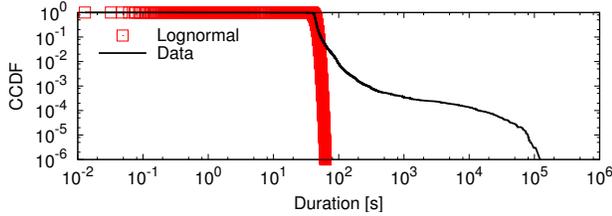
This work was supported by the MIUR-PRIN funded project “FAMOUS Fluid Analytical Models Of aUtonomic Systems” and by the MIUR-PRIN funded project “RECIPE



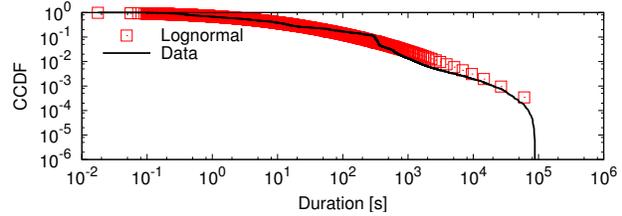
(a) eDonkey - download;  $\mu = 5.05$ ;  $\sigma = 0.72$



(b) BitTorrent - download;  $\mu = 5.97$ ;  $\sigma = 0.93$

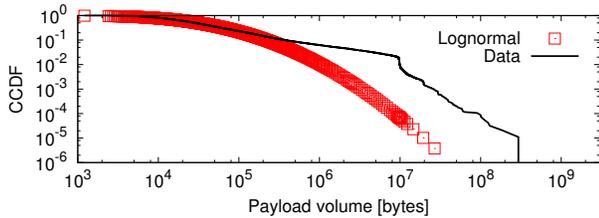


(c) eDonkey - signaling;  $\mu = 3.79$ ;  $\sigma = 0.07$

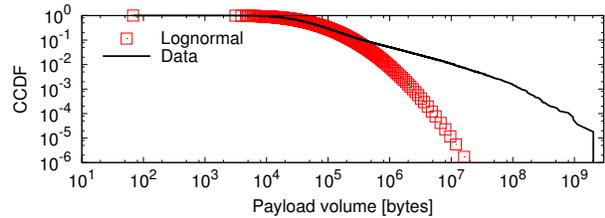


(d) BitTorrent - signaling;  $\mu = 1.47$ ;  $\sigma = 2.81$

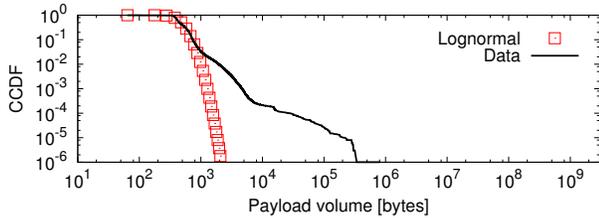
**Figure 3. CCDF of the conversation durations.  $\mu$  and  $\sigma$  are the lognormal parameters**



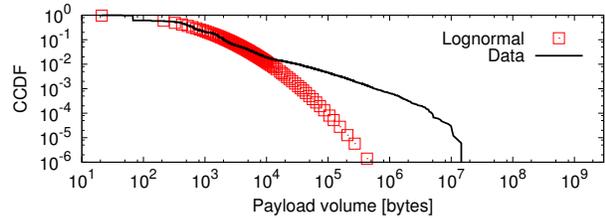
(a) eDonkey - download;  $\mu = 10.35$ ;  $\sigma = 1.51$



(b) BitTorrent - download;  $\mu = 10.86$ ;  $\sigma = 1.24$



(c) eDonkey - signaling;  $\mu = 6.18$ ;  $\sigma = 0.32$



(d) BitTorrent - signaling;  $\mu = 5.73$ ;  $\sigma = 1.54$

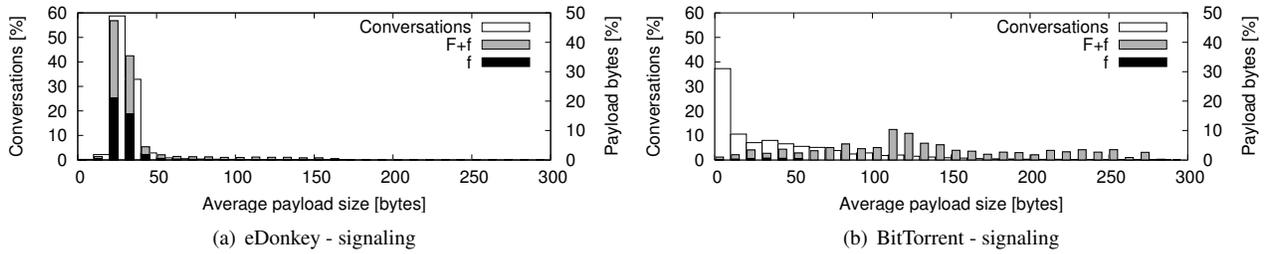
**Figure 4. CCDF of the conversation volumes.  $\mu$  and  $\sigma$  are the lognormal parameters**

Robust and Efficient traffic Classification in IP nEtworks”.

We would like to thank CSITA for supporting us in this research.

## References

- [1] R. Bolla, M. Canini, R. Rapuzzi, and M. Sciuto. Characterizing the network behavior of P2P traffic. In *Proceedings of 4th International Telecommunication Networking Workshop on QoS in Multiservice IP Networks (QoS-IP)*, Feb. 2008. to appear.
- [2] D. Erman, D. Ilie, and A. Popescu. Bittorrent session characteristics and models. In *HET-NETs '05 Conference Proceedings*, July 2005.
- [3] L. Guo, S. Chen, Z. Xiao, E. Tan, X. Ding, and X. Zhang. Measurements, analysis, and modeling of bittorrent-like systems. In *Proceedings of Internet Measurement Conference*, pages 35–48, Oct. 2005.
- [4] IPP2P. <http://www.ipp2p.org>.
- [5] T. Karagiannis, K. Papagiannaki, and M. Faloutsos. BLINC: Multilevel traffic classification in the dark. In *Proceedings of ACM Sigcomm*, Aug. 2005.
- [6] E. Kohler. Click for measurement. Technical Report TR060010, UCLA Computer Science Department, Feb. 2006.
- [7] E. Kohler, R. Morris, B. Chen, J. Jannotti, and M. F. Kaashoek. The click modular router. *ACM Transactions on Computer Systems*, 18(3):263–297, Aug. 2000.



**Figure 5. Relationships between the conversations, volumes and the average payload size. The histograms are plotted with bin size 10 bytes**

- [8] R. Pang, V. Paxson, R. Sommer, and L. Peterson. binpac: A yacc for writing application protocol parsers. In *Proceedings of ACM Sigcomm Internet Measurement Conference*, Oct. 2006.
- [9] S. Sen, O. Spatscheck, and D. Wang. Accurate, scalable in-network identification of P2P traffic using application signatures. In *Proceedings of the 13th international conference on World Wide Web*, May 2004.
- [10] K. Tutschku. A measurement-based traffic profile of the edonkey filesharing service. In *Proceedings of PAM*, Apr. 2004.