EURECOM
Sophia Antipolis

Institut Eurécom
Department of Corporate Communications
2229, route des Crètes
B.P. 193
06904 Sophia-Antipolis
FRANCE

Research Report RR-06-178

Application-Level Performance of ADSL Users

11th October 2006

Matti Siekkinen[1], Denis Collange[2], Guillaume Urvoy-Keller, Ernst W. Biersack

Tel : (+33) 4 93 00 26 26
Fax : (+33) 4 93 00 26 27
Email : siekkine@ifi.uio.no, denis.collange@orange-ft.com,
{urvoy,erbi}@eurecom.fr

[1]University of Oslo, Department of Informatics, Postbox 1080 Blindern, 0316 Oslo, Norway
[2]France Télécom R&D, 905, rue Albert Einstein, 06921 Sophia-Antipolis, France

[0]Institut Eurécom's research is partially supported by its industrial members: Bouygues Téécom,Fondation d'entreprise Groupe Cegetel, Fondation Hasler, France Télécom, Hitachi, ST Microelectronics, Swisscom, Texas Instruments, Thales

Abstract

We report results from the analysis of a 24-hour packet trace containing aggregate TCP traffic of approximately 1300 residential ADSL clients. We make observations similar to earlier studies by the research community: the major fraction of the total traffic originates from P2P applications and small fractions of connections and clients are responsible for the vast majority of the traffic. However, our main contribution is throughput performance analysis of the clients. We observe suprisingly low utilizations of upload and download capacity for most of the clients. Furthermore, by using our TCP root cause analysis tool, we obtain a striking result: in over 90% of the cases, the low utilization is mostly due to the applications that clients use and that limit the transmission rate and not network congestion, for instance. P2P applications typically impose upload rate limits to avoid uplink saturation that would damage download performance. Our analysis shows evidence that these rate limits, set either by the user or by the application, are too conservative and, as a consequence, the overall performance of these applications is poor. Deployment of more intelligent rate limit mechanisms in such a scenario would imply increase in clients' throughput at the expense of an increased load in backbone networks.

1 Introduction

We analyze a large packet trace of clients connected to the Internet via ADSL. We focus on the performance limitations from a client point of view and and use a TCP root cause analysis tool that we apply to TCP connections. We consider throughput as the performance metric. The cause that limits the performance of a particular connection can be located either at the edge (sender or receiver) of a connection or inside the network. Limitations at edge comprise the application not providing data fast enough to the TCP sender or the TCP receiver window being too small. A network limitation results from the presence of a bottleneck that can be anywhere along the end-to-end path. We perform root cause analysis of performance both, at connection level and at client level. Based on a packet level trace that captures the activity of over one thousand ADSL clients during 24 hours we see that

- The distribution of the client activity in terms of volume (resp. duration) is highly skewed (resp. peaked). Most clients are active only during a short period of time. Also, most clients generate a limited amount of traffic in the order of several MBytes, while a small number of (heavy hitter) clients upload and download hundreds of MBytes each.

- The utilization of the uplink and downlink is very low for most of the clients. Even heavy hitters are far from saturating their access link.

- The low utilization is mainly due to the applications that limit their rate of transfer, which is now very common for P2P applications such as eDonkey.

The rest of the paper is organized as follows. We first describe in Section 2 the measurement setup . Then, in Section 3, we look at some general characteristics of the traffic and clients' behavior. In Section 4, we focus on the performance analysis of the clients before we conclude the paper.

2 Architecture and Setup

The ADSL architecture is organized as follows (see Figure 1): the Broadband Access Server (BAS) aggregates the traffic issued from many Digital Subscriber Line Access Multiplexer (DSLAM) before forwarding it through the two local routers to an IP backbone. Each client is connected to one DSLAM using one ATM Virtual Circuit. The traffic of a client is controlled by the up and down capacities of this access link. A variety of subscription types of the ADSL access service is defined through different combinations of uplink and downlink capacities.

Traffic is captured using two probes. Those probes are located between a BAS and the first two routers of the IP backbone. Each probe captures packets flowing through a single router. This BAS multiplexes the traffic of three DSLAMs. It connects around 3000 clients to the Internet. We capture all IP, TCP and UDP

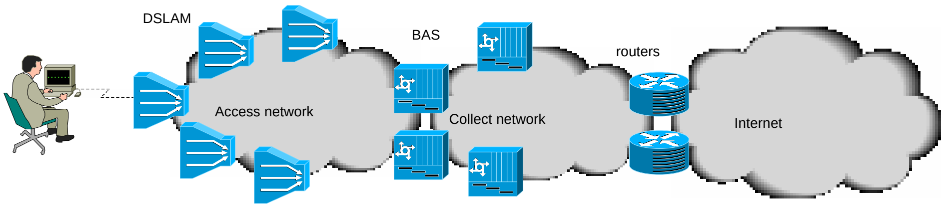


Figure 1: Architecture of the ADSL platform monitored.

headers of packets going through the BAS without any sampling or loss. Trace files have a tcpdump format.

Our analysis is based on a whole day of traffic measurements, Friday March 10, 2006. The data collected on this day represents approximately 290 GB of TCP traffic in total out of which 64% is downstream and 36% upstream. According to observations on other days, these volumes are representative of a typical day's activity on this network.

In addition to the packet trace file, we have a list of IP addresses that belong to local clients, which allows us to distinguish the direction of the traffic.However, we do not know the clients subscription rates, i.e. their uplink and downlink capacities.

3 Traffic Characteristics: Applications, Connections, and Clients

3.1 General Characteristics of the Traffic

3.1.1 Traffic per Application

Figure 2 shows how the amount of bytes transferred evolves and is distributed between the most common applications for each half an hour period. Bytes transferred upstream are on the positive part of the y-axis and bytes transferred downstream on the negative part of the y-axis. We account separately for those applications that generated more than 5% of the total amount of bytes. Those applications are only five (email comprises SMTP, POP3, and IMAP traffic). Remaining applications are included in the"other" category. We associated the TCP port range 4660-4669 to eDonkey, the ports 6880-6889 and 6969 (tracker) to BitTorrent, and standard TCP port numbers for the rest of the applications.

The application responsible for most of the transferred bytes is eDonkey followed by traffic originating or destined to ports 80 and 8080. We do not want to declare the traffic seen on ports 80 and 8080 as Web traffic since it is likely to include also P2P traffic, as we will see later. The dominant category of traffic, however, is the "other" traffic. Since much of todays traffic is not using fixed ports but "hiding" [4], we are not able with our port-based method to classify much of

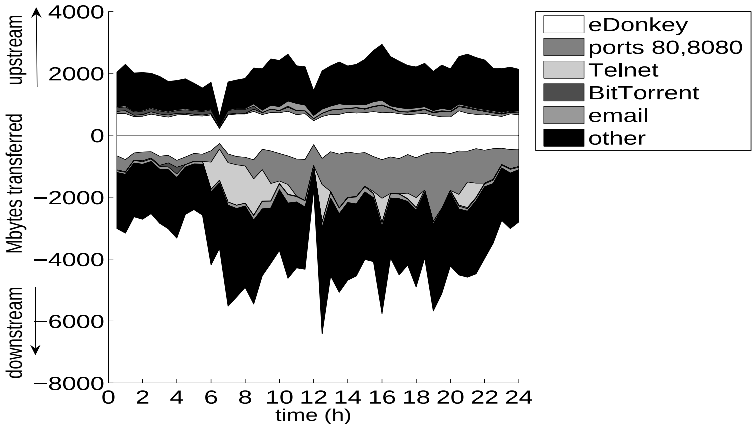


Figure 2: Amount of bytes transferred by different applications during the day.

the traffic seen. Therefore, the "other" traffic represents about 50% of the total traffic. We plan to use more advanced techniques [4] in the future to identify more applications.

The amount of eDonkey traffic is almost constant throughout the 24 hours. The fact that the traffic of the "other" applications stays also almost constant throughout the 24 hours can be considered as an indication that this class contains P2P traffic. For port 80/8080 traffic we observe a diurnal pattern: The traffic is almost negligible in the middle of the night and hits its peak at 2pm and 7pm. This effect is much more pronounced in the downstream traffic. Interestingly, BitTorrent traffic appears only during the night and almost only upstream. This observation suggests that there is some hidden P2P at least in the downstream traffic since it is very unlikely that BitTorrent clients would only upload data. As one could expect, email traffic is more present during the day than during the night. Telnet traffic emerges from time to time in an inexplicable way. We looked at this traffic in more detail and it turned out to consist of only few long and fast transfers originating from a couple of hosts.

3.1.2 Traffic per Connection

Let us now focus on individual TCP connections, which we will analyze in more detail in Section 4. TCP connections were identified through unique four tuples that consist of source and destination IP addresses and TCP port numbers. Figure 3(a) shows the complementary cumulative distribution function (CCDF) of the connection sizes. We see that the connection sizes span a wide range of values with the largest ones being in the order of several tens of Megabytes. In addition, while most of the connections are very small they do not, as Figure 3(b) shows, contribute much to the total traffic. Those results are in line with the ones observed since a decade, especially on Web traffic [2].

3.2 Client Behavior

While we observed packets from approximately 3000 clients, our analysis focuses only on a subset of them: those 1335 clients that generated at least one long enough connection for root cause analysis (see Section 4.1).

3.2.1 Volumes and Applications

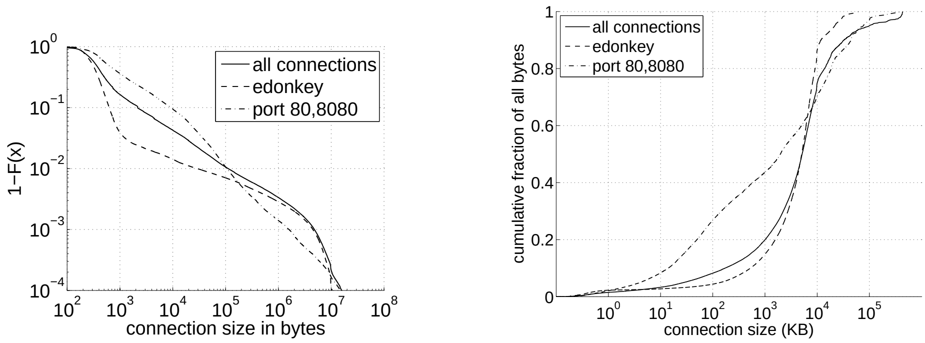
We focus here on the client behavior in terms of volumes of data transferred and applications used. Figure 4(a) shows the distribution of the bytes transferred by clients, and Figure 4(b) shows the cumulative fraction of bytes contributed by a client that transferred a given amount of bytes. These two figures tell us that the amount of traffic generated per client is heavily skewed: About 15% of the most active clients transfer roughly 85-90% of the total bytes, both upstream and downstream. These 15% account for 200 out of the total of 1335 analyzed clients and we refer to them as the **heavy-hitters**. A study that was recently performed on a much larger scale for Japan's residential user traffic [1] reported that 4% of heavy-hitter clients s account for 75% of the in-bound and 60% of the out-bound traffic.

Note that these two sets of heavy-hitter clients, upstream and downstream, are distinct sets that both comprise about 200 clients. However, the sets are heavily overlapping since among these 200 clients, 128 clients are in both sets, which indicates that the majority of the heavy-hitters both, upload and download a lot of data, which comes most likely from P2P applications. The average amount of bytes uploaded and downloaded by a heavy-hitter client is approximately 470 MB and 760 MB, respectively, while for the non-heavy-hitters these average values are 9 MB and 27 MB. We also looked at the duration these heavy-hitter clients are active during the day and found two groups: One group that is active between 30 minutes and 2 hours, and a second much smaller group that is active up to 24 hours.

We next compare the profile of an average client and of an average heavy hitter in terms of the applications they use. To do so, we compute for both groups how much certain applications contribute to the total amount of traffic. Then, we selected for each client the application that generated the most bytes. The results are shown in Table 1. The main difference between an average client and a heavy hitter is that heavy hitters tend to use P2P applications (esp. eDonkey) more extensively.

3.2.2 Access Link Utilization

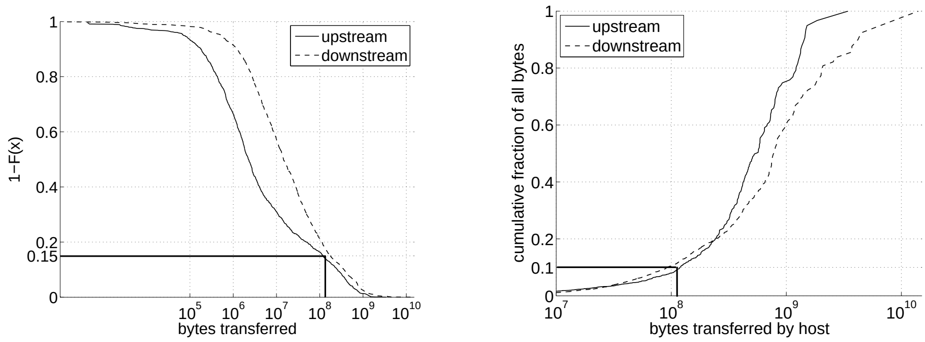
To compute the utilization of a link, one needs to know its capacity. As we do not have this knowledge, we need to approximate it by using an estimate. Note that we cannot use tools such as Pathrate [3] that estimate capacity of an entire path, which is equal to the capacity of the link with the smallest capacity along the path, because the local access link is in many cases not the one with the smallest capacity on the path. For instance, in P2P download from another ADSL client, the downlink capacity of the local client is very likely to be higher than the uplink



(a) CCDF plot of connection size. Note the logarithmic scale of both axes.

(b) Cumulative fraction of all bytes as a function of the connection size.

Figure 3: Connection statistics.



(a) CCDF plot of bytes transferred by clients.

(b) Cumulative fraction of all bytes transferred as a function of bytes transferred by a given client.

Figure 4: Volume per client.

Table 1: Percentages of clients that transmit most bytes using a specific application.

Upstream						
	total	**eDonkey**	**Ports 80,8080**	**BitTorrent**	**email**	**others**
non-heavy-hitters	1135	6.1%	4.1%	0%	0.53%	89%
heavy hitters	200	44%	2.5%	0.50%	0.50%	53%
Downstream						
	total	**eDonkey**	**Ports 80,8080**	**BitTorrent**	**email**	**other**
non-heavy-hitters	1135	8.4%	6.9%	0%	0.26%	84%
heavy hitters	200	28%	9.5%	0%	1.0%	62%

capacity of the distant peer. As estimate, we use the *maximum observed instantaneous throughput*, which gives a lower bound for the access link capacity. The instantaneous throughput for each client is computed over non-overlapping intervals of five seconds. Since we select for each client the 30-minute period where the instantaneous throughput seen by that client was highest during the whole day, we increase the chances of obtaining an estimate of the capacity that is closer to the true value. However, the period of the day used to estimate the capacity for each client is not necessarily the same for all clients.

For each client, we compute the link utilization for that period of 30 minutes during which the client achieved its highest instantaneous throughput. If we define the mean aggregate throughput of a client as the total amount of bytes uploaded or downloaded by that client during 30 minutes divided by 30 minutes, we can compute its **utilization** as mean aggregate throughput divided by maximum instantaneous throughput. In this way, we obtain an *upper bound for the utilization*, because we use a lower bound for the estimate of the capacity.

Figure 5 shows the CDF plot of the utilization. we see that overwhelming majority of clients are far from fully utilizing their access links. This is even more the case if we remember that our approximation of the utilization tends to *over estimate* the actual utilization. We see that 80% of the clients have an utilization of less than 20% for their downlink and less than 40% for their uplink.

Having seen that most clients achieve very low link utilization, we will now set out to investigate the causes. For this purpose, we will use some techniques referred to as root cause analysis (RCA) that has been originally proposed by Zhang et al. [9] and further refined by Siekkinen et al. [8].

4 Performance Analysis of Clients

4.1 Connection-Level Root Cause Analysis

To apply RCA, we need TCP connections that carry at least 130 data packets, which is equivalent to about 190 KB of data, if we assume MSS to be 1450 Bytes.

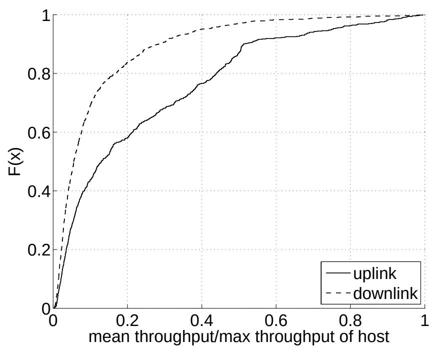


Figure 5: CDF plot of upper bound for link utilization per client for a 30 min period. For each client, we selected the period during which that client achieved its maximum instantaneous throughput.

From Figure 3 we know that most connections are quite small, but that most of the bytes are carried in a tiny fraction of the largest connections. As a consequence, our RCA will only be able to analyze the 1% of the largest connections, which however carry more than 85% of total bytes.

Identifying the factors (causes) that limit the throughput achieved by a TCP connection is not an easy task. Also, for very long connections, different causes can intervene at different points in the lifetime of a connection. For this reason, we classify in a first step the packets of a connection into two groups. Each packet is either part of an **application limited period** (ALP) or a **bulk data transfer period** (BTP).

Roughly speaking, the throughput of packets that are part of an ALP is *limited by the behavior of the application*. For example, an IP telephony application that produces packets at a fixed rate clearly determines (and limits) the throughput achieved. Therefore, the packets of the TCP connection carrying these data should all be put into an ALP. The packets that are not part of an ALP will be part of a BTP. For the details on how packets get classified into ALPs and BTPs, we refer to our technical report [6].

For packets that are part of a BTP, there can be a number of causes that limit the throughput achieved, such as

- **Network limitation**. This limitation corresponds to the case where a bottleneck limits the observed throughput. We distinguish between two types of network limitation. One is referred to as **un-shared bottleneck** and corresponds to the case where a single connection uses the full capacity of the bottleneck link, as compared to the second case, referred to as **shared bottleneck** where several connections share the capacity of the bottleneck link.

- **TCP end-point limitation**. This limitation corresponds to the case where the advertised receiver window is too small as compared to the bandwidthdelay product of the path, which prevents the sender to achieve a higher throughput.

The decision has to which limitation is the more likely is based on a set of time series-based metrics computed from the packet header trace of the connection and a threshold-based classification scheme. For details, we refer the reader to [7, 8].

4.2 Client-Level Root Cause Analysis

We are interested in doing RCA not only at connection level but also at *client* level. We identify four types of limitations for clients, which are: (i) Applications, (ii) Access link saturation, (iii) Network limitation due to a distant bottleneck, and (iv) TCP end-point limitation. TCP end-point limitation is due to unnecessarily small receiver advertized window values that prevent TCP connections from saturating the path. Our analysis showed this cause to be marginal in our data set. Hence, we exclude this limitation from further discussions.

In this analysis, we focus on *active* clients. We define a client to be *active* during a period of 30 minutes if it transferred at least 100 KB during that period. For each active client we consider all the bytes transferred by all the connections of the client within a given 30-minute period. We then associate these bytes into the three considered client-level limitations. To do this association, we use the connection-level RCA as follows: All the bytes carried by the ALPs of all the connections of the client are associated to application limitation. All the bytes carried by all the BTPs that are labeled network limited (unshared or shared bottleneck) by connection-level RCA and during which the utilization is above 90% of the maximum are associated to access link saturation. All the bytes carried by the rest of the network limited BTPs during which the utilization is below 90% of the maximum are associated to network limitation due to a distant bottleneck. All the rest of the bytes transferred by the client, and not covered by these three limitations, are associated to "other" (unknown) client limitation. The amount of bytes associated with each limitation serves as a quantitative metric of the degree of that limitation for a given client during a given 30-minute period.

We know from our previous work on RCA that for a single, possibly very long connection, the limitation cause may vary over time. Also, a single client may run one or more applications that will originate multiple connections. Assigning a single limitation cause to each client is therefore tricky. For this reason, we distinguish for each client between "main limitation" and "limitations experienced". As **main limitation**, we understand the limitation that effects the most number of bytes for this client. This classification is exclusive. i.e. each client belongs to a single limitation category.

On the other hand, under **limitations experienced** a single client will be considered in all the categories whose limitation causes it has experienced. Therefore, this classification is not exclusive. The results are presented in Table 2. We present the results for two 30-minute periods of the day: 4-4:30am and 3-3:30pm, which are representative for the different periods of the day. We see that during the night time heavy hitters dominate (70 out of 77 active uploading clients and 61 out of 83 active downloading clients), which is not surprising if one considers that heavy hitters heavily use P2P applications and P2P file transfer that can run for several hours [5]. If we look at the absolute number of clients, we see that only a small fraction of 1335 clients is active in either 30-minute period.

4.2.1 Main Limitation

If we look at the main limitation cause experienced by the clients, we see that *almost all clients see their throughput performance mainly limited by the application*. This holds irrespectively of the direction of the stream (upstream or downstream), of the type of client, average client or heavy hitter, and of the period of the day.

The clients that are not application limited see their throughput either limited by the capacity of the access link or the capacity of another link along the end-to-end path. Capacity limitations occur more frequently during the daytime than at night. The very limited number of cases where we observe a saturation of the access link complies with the low utilization observed in Figure 5.

4.2.2 Limitations Experienced

Besides the main limitation, we also consider *all the limitation causes* experienced by a single client. The most striking result is the difference between main limitation and limitations experienced for the ”other link” limitation. As we have seen, this limitation is rarely the main limitation, while the percentage of clients that experience such limitation is between 40% and 60%.

4.3 Throughput limitations causes experienced by major applications

Having done the root cause analysis on a per-client basis, we now investigate what are the most important applications that experience the different limitation causes, namely (i) application limited, (ii) saturated access link, and (iii) bottleneck at distant link.

Figure 6(a) shows the main applications that generate traffic that is application limited. We compute the different amounts by simply summing all the bytes for all the ALPs for each 30 minute period. If we look at the evolution of the total volume of traffic that is application limited we see very little variation in time and an upload volume almost as big as the download volume, both being around 2 GBytes per 30 minutes. The largest single application that generates application limited traffic is, as expected, eDonkey. However, if we look by volume, the largest category is “other”, i.e. the one where we were not able to identify the application generating the traffic. The overall symmetry of upload and download volumes for the “other” category as well as a manual analysis of the traffic of some heavy hitters strongly suggest that the ”other” category contains of a significant fraction of P2P traffic.

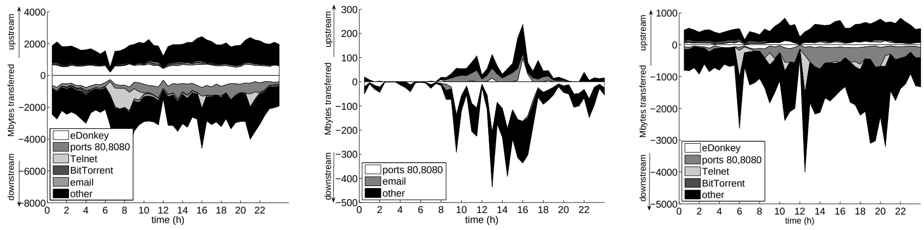
Table 2: Number of active clients limited by different causes.

Upstream							
limitation cause			*Total active #*	*application*	*access link*	*other link*	*other cause*
main limitation	all clients	4am	77	95%	0%	4%	1%
		3pm	205	86%	6%	4%	4%
	heavy hitters	4am	70	94%	0%	4%	2%
		3pm	111	92%	2%	3%	3%
limitations experienced	all clients	4am	77	100%	0%	60%	–
		3pm	205	100%	7%	39%	–
	heavy hitters	4am	70	90%	0%	66%	–
		3pm	111	92%	5%	64%	–
Downstream							
limitation cause			*Total #*	*application*	*access link*	*other link*	*other cause*
main limitation	all clients	4am	83	93%	1%	4%	2%
		3pm	286	76%	4%	18%	2%
	heavy hitters	4am	61	97%	0%	2%	1%
		3pm	114	80%	2%	16%	2%
limitations experienced	all clients	4am	83	100%	1%	53%	–
		3pm	286	100%	7%	42%	–
	heavy hitters	4am	61	100%	0%	59%	–
		3pm	114	100%	4%	61%	–

Figure 6(b) shows the main applications that saturate the access link. For this cause, no traffic originating from recognized P2P applications was seen. Instead, a significant portion of traffic saturating the uplink is e-mail. For the downlink it is mainly traffic on ports 80 and 8080 and traffic for which the application could not be identified. The fact that the traffic using ports 80 and 8080 primarily saturates only downlink suggests that it could be real Web traffic that consists of small upstream requests and larger downstream replies from the server, as opposed to P2P traffic which is typically more symmetric. If we compare the absolute volume we see that most of the activity is concentrated to day time, with the peak being in the early afternoon and a total volume that is even at its peak almost negligible as compared to the traffic volume that is application limited (see Figure 6(a)).

Figure 6(c) shows the main applications that see their throughput limited by a link that is not the access link. Here, the category of other applications is clearly dominating in terms of volume. Otherwise, we observe a mixture of applications. It is expected that the set of applications is diverse since this type of network limitation can occur at any point of the network regardless of the application behavior at the client side.

In the download direction, the total traffic that is limited by a distant bottleneck reaches in the late afternoon a proportion that, in terms of volume, is almost as important as the download traffic that is application limited. The fact that this traffic



(a) Application limitation.

(b) Access link saturation.

(c) Network limitation due to a distant bottleneck.

Figure 6: Amount of bytes experiencing a particular root cause.

peaks late afternoon[3] may be an indication of higher overall network utilization just after working hours, not only within the France Telecom network but in wider scale, that causes more cross traffic in aggregating links. Note that at the same time, the amount of traffic limited by the access link is very low (Figure 6(b)).

Finally, we would like to point out that a comparison of the absolute traffic volumes of Figures 6(a) – 6(c) reveal that the application limitation category represents almost 80% of the total number of transmitted bytes.

4.3.1 Root Causes for main applications

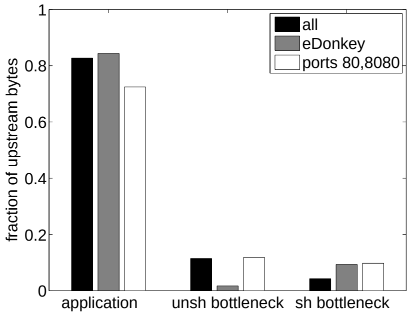
So far we concentrated on per-client analysis of the causes. However, it is interesting to do a root cause analysis per application class. In Figure 7 we can see that application limitation is clearly the dominant limitation cause. This is not surprising given that we already saw that a vast majority of the heavy hitter clients experience their throughput to be application limited. For the traffic that is not application limited, network limitation is the major root cause.

4.4 Impact of the Root Causes on Link Utilization

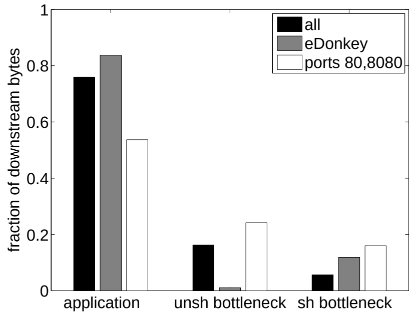
We have seen that for the traffic we study there are three main root causes that limit the throughput. We now want to know how these root causes impact the link utilization of the clients. We focus on link utilization and not on absolute throughput, because clients have different link capacities and we want to understand how far we are from the maximum utilization, i.e. link saturation. Intuitively, saturation is likely to be reached for the case of limitation by saturated access link.

As before, we included in the analysis for each client only the traffic of the 30-minute period for which that client achieved its highest instantaneous throughput. We computed client's link utilization during ALPs and BTPs limited by different

[3] An analysis of the IP addresses using Maxmind (http://www.maxmind.com/) revealed that most of the local clients exchange data primarily with peers/servers located in France or surrounding countries.



(a) upstream



(b) downstream

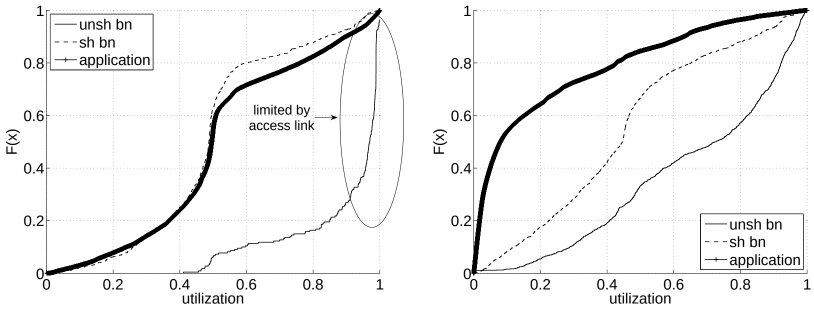
Figure 7: Major limitation causes for different application classes.

causes. In this way, we can quantify the impact of different limitation causes on the performance. Figure 8 shows CDF plots of the results.

We focus first on uplink utilization: We see that for the case of an unshared bottleneck, the utilization is in approximately 70% of the cases very close to one, which means that in these cases the uplink of the client is the bottleneck. In the remaining 30% of cases where we observe an unshared bottleneck, we see a link utilization between 0.4 and 0.85 that can be due to a distant access *down*-link, e.g. a peer that has lower downlink capacity than the uplink capacity of the local peer, or due to simply misclassification. For the two other root causes, application limitation and shared bottleneck, the clients achieve in about 60% of the cases a link utilization of less than half the uplink capacity.

If we look at the utilization of the downlink, we see that application limited traffic results most of the time in a very poor downlink utilization. Given that most of the application limited traffic is eDonkey traffic (cf. Figure 6(a)), one might be tempted to explain this low utilization by that fact that most likely the peer that sources the data has an asymmetric connection with the uplink capacity being much lower than the downlink capacity of the receiving peer[4]. However, a downloading peer has usually multiple parallel download connections, which in aggregation should be able to fully utilize the downlink capacity. The fact that this is not the case seems to indicate that many users of eDonkey use the possibility to rate-limit their upload rate to a rate much lower than the capacity of their uplink. Figure 9, which plots the maximum instantaneous aggregate download rates achieved per-client for different applications, further underlines this effect. We see that the maximum aggregate download rates of P2P applications, eDonkey and BitTorrent, fall clearly behind the maximum download rates of FTP and port 80/8080

[4]Maxmind also reported that a clear majority of the distant IPs that the heavy-hitters communicated with were clients of ISPs providing residential services.



(a) upstream (b) downstream

Figure 8: CDF plot of access link utilization for the different root causes. For each client, we consider only traffic of the 30 min period during which that client achieved the highest instantaneous throughput of the day.

traffic. A recent study of eDonkey transfers by ADSL clients [5] found that the average file download speed achieved was only a few KByte/sec. Our findings seem to indicate that such a poor performance is not due to network or access link saturation but rather due to *eDonkey users drastically limiting the upload rate of their application.*

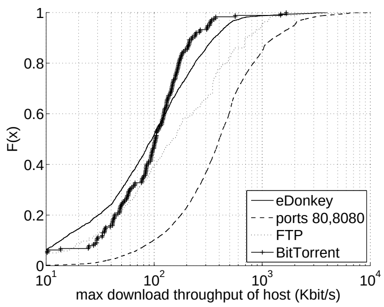


Figure 9: CDF plot of maximum aggregate per-host download throughput computed over five second intervals.

4.5 Comparison With Other Related Analysis Work

In [9], Zhang et al. performed flow-level root cause analysis of TCP throughput. They analyzed packet traces collected at high speed access links connecting two sites to the Internet; a peering link between two Tier 1 providers; and two sites on a backbone network. As results, the authors reported that, in terms of traffic volumes, congestion (similar to network limitation in our vocabulary) was the most common limiting factor followed by host window limitation (TCP end-point in our vocabulary). It is important to notice that their studies were based on data collected in 2001-2002. At that time, the popularity of P2P applications such as eDonkey was far from what it is today. We refer the reader to [6–8] for details about differences between our RCA tools and their T-RAT.

In order to understand if our results are specific to this particular access network, we applied our RCA tool also to other publicly available packet traces collected at an ADSL access network in Netherlands (http://m2c-a.cs.utwente.nl/repository/). We looked at two 15-minute traces: one captured in 2002 and another one in 2004. A similar port based study than in Section 3.1 showed that in the 2002 trace, the applications generating most traffic where FTP and applications using ports 80 and 8080, while eDonkey and BitTorrent were dominating in the 2004 trace. We were unable to perform similar client-level study due to lack of knowledge about local client IP addresses and limited capture durations. However, simple connection-level RCA revealed that in the 2002 trace around 40% of bytes were throughput limited by the application. In the 2004 trace, this percentage was already roughly 65%, which demonstrates the impact of the increase in P2P application traffic.

5 Conclusions

We presented an analysis of one day ADSL traffic generated by more than one thousand clients. Some of our findings corroborate the results of earlier studies: We saw that (i) a major fraction of the total traffic is P2P traffic whos volume varies very little over the course of a day and (ii) the connection sizes as well as the amount of bytes generated per client are very skewed, with few connections (resp. clients) being responsible for the majority of the traffic.

The other findings, however, are quite surprising and we have not seen them presented elsewhere. In particular, we observed that most of the clients never use more than a very small fraction of the upload and download capacity. TCP root cause analysis revealed that most of the user traffic is in fact application limited, which means that the users of P2P applications impose upload rate limits that are chosen to be very low. Other root causes that were typically observed in other packet traces [9] play a only a minor role: We saw some rare occurrences of network limitation but we did not see any throughput limitation due TCP configuration issues such as too small a receiver window.

By severely limiting the aggregate upload rate of their P2P applications, the clients certainly make sure that their P2P traffic does not interfere with concurrent activities such as Web serving or IP telephony. However, this comes at the price of necessarily high download times as a result of poor utilization of the network bandwidth. We therefore conclude that the current rate limitation strategies used by P2P clients is very inefficient from a users point of view.

The implication of such a low access link utilization is naturally low utilization of the entire access network, which is beneficial for the service provider. However, there is a caveat: the utilization and, consequently, the traffic volumes can change dramatically in case a new type of popular P2P application is deployed or an already one is upgraded to utilize the uplink in a different, more intelligent way.

Our work is by no means finished. These interesting insights provide vital guidance for future directions. For instance, it would be interesting to perform a similar study that spans over several days, study in more detail the role of different applications, esp. eDonkey, as sources of performance limitations, and perform more detailed profiling of clients.

Acknowledgments

This work has been partly supported by France Telecom, project CRE-46126878.

References

[1] K. Cho, K. Fukuda, H. Esaki, and A. Kato, "The Impact and Implications of the Growth in Residential User-to-User Traffic", In *Proceedings of ACM SIGCOMM '06*, pp. 207–218, New York, NY, USA, September 2006, ACM Press.

[2] M. E. Crovella and A. Bestavros, "Self-similarity in World Wide Web traffic: evidence and possible causes", *IEEE/ACM Transactions on Networking*, 5(6):835–846, 1997.

[3] C. Dovrolis, P. Ramanathan, and D. Moore, "Packet-dispersion techniques and a capacity-estimation methodology", *IEEE/ACM Trans. Netw.*, 12(6):963–977, 2004.

[4] T. Karagiannis et al., "Is P2P dying or just hiding?", In *Proc. Globecom 200*, November 2004.

[5] L. Plissonneau, J.-L. Costeux, and P. Brown, "Detailed Analysis of eDonkey Transfers on ADSL", In *Proceedings of EuroNGI*, 2006.

[6] M. Siekkinen, G. Urvoy-Keller, and E. W. Biersack, "On the Impact of Applications on TCP Transfers", Technical Report, Institut Eurecom, October 2005, http://www.eurecom.fr/˜siekkine/pub/RR-05-147.pdf.

[7] M. Siekkinen, *Root Cause Analysis of TCP Throughput: Methodology, Techniques, and Applications*, Ph.D. Thesis, Institut Eurécom / Université de Nice-Sophia Antipolis, Sophia Antipolis, France (http://www.ifi.uio.no/˜siekkine/pub.html), October 2006.

[8] M. Siekkinen, G. Urvoy-Keller, E. Biersack, and T. En-Najjary, “Root Cause Analysis for Long-Lived TCP Connections”, In *Proceedings of CoNEXT*, October 2005.

[9] Y. Zhang, L. Breslau, V. Paxson, and S. Shenker, “On the Characteristics and Origins of Internet Flow Rates”, In *Proceedings of ACM SIGCOMM ’02*, Pittsburgh, PA, USA, August 2002.