Analysis of Internet Services in IP over ATM networks

J. Aracil, D. Morató and M. Izal
Dept. de Automática y Computación
Universidad Pública de Navarra
Campus Arrosadía s/n
31006 Pamplona, SPAIN
Tel: +34 948 16 97 33
Fax: +34 948 16 92 81
email: javier.aracil@upna.es, daniel.morato@upna.es, mikel.izal@upna.es

March 29, 1999

Abstract

This paper presents a trace-driven analysis of IP over ATM services from a user-perceived quality of service standpoint. QoS parameters such as the sustained throughput for transactional services and other ATM layer parameters such as the burstiness (MBS) per connection are derived. On the other hand, a macroscopic analysis that comprises percentage of flows and bytes per service, TCP transaction duration and mean bytes transferred in both ways is also presented. The traffic trace is obtained with a novel measurement equipment that combines a header extraction hardware and a high end UNIX workstation capable of providing a timestamp accuracy in the order of microseconds. The ATM link under analysis concentrates traffic from a large population of 1,500 hosts from Public University of Navarra campus network, that produce 1,700,000 TCP connections approximately in the measurement period of one week. The results obtained from such a wealth of data suggest that QoS is primarily determined by transport protocols and not by ATM bandwidth. The sustained throughput of TCP connections never grows beyond 80 Kbps with 70% probability in the data transfer phase (i. e., in the ESTABLISHED state) and we observe a strong influence of the connection establishment phase in the user-perceived throughput. On the other hand, the burstiness of individual TCP connections is rather small, namely TCP connections do not produce bursts according to the geometric law given by slow start and commonly assumed in previously published studies.

Keywords: Internet services, quality of service, IP over ATM, traffic measurements.

1 Introduction

The increasing demand for Internet services is sparking the deployment of high-speed IP backbones but the specific technology to be used in such high-speed backbones still remains an open issue. One of the main advantages of ATM networks is the capability to provide dynamic resource allocation for video, voice and data, thus making ATM ideally suited for the high variability of IP traffic. However, the state of the art in ATM equipment reveals that only limited capabilities are provided for such dynamic resource allocation. Indeed, the ATM technology readily available in the telecommunication market is able to provide constant bit rate (CBR) and best effort (Unspecified Bit Rate - UBR-) Virtual Circuits (VCs) only. The former service class offers static allocation of resources while the latter serves to the purpose of statistical multiplexing of ATM cells, which are filtered by a traffic shaper prior to entering the VC. Since the best-effort paradigm is commonly used for the most practical implementations of IP over ATM links, we are currently witnessing a number of alternatives to ATM, like for example IP over WDM [1], that promise best-effort service with no additional segmentation and reassembly overhead. Thus, it becomes clear that the success of ATM as the Next Generation Internet (NGI) backbone technology will largely depend on its ability to offer not only a best effort service but end-to-end QoS to the telecommunication services user.

Nevertheless, the provision of end-to-end QoS for IP over ATM networks is a rather involved issue, since QoS is not totally under the network control. For example, the most demanded Internet services (WWW, email, FTP) use TCP as a transport protocol. Thus, QoS is not only determined by network capacity but the TCP dynamics are also of primary importance. A careful examination of the quality of service for Internet streams, that takes into account the contributions of the transport protocol and the underlying ATM network parameters is thus in order. Precisely, the aim of this paper is to provide an IP services analysis in the context of an IP-over-ATM access link to the Internet. Such scenario is common for Internet service providers and corporate networks that are connected to the Internet using ATM technology. The goal of the analysis is twofold:

- First, we present the macroscopic characteristics of the IP-over-ATM stream, in terms of percentage of flows and bytes per service, TCP transaction duration and bytes.

- Secondly, we focus on user-perceived quality of service for the most popular Internet services, and relate to transport protocol dynamics and ATM traffic descriptors in order to provide insight on the actions to be performed at the ATM layer to enhance user-perceived quality of service.

Since TCP behavior is a relevant factor to user-perceived quality of service, irrespective of the underlying network technology, we note that a number of papers have appeared in the recent technical literature. In order to evaluate to what extent the results of such papers are applicable to our case study and to gain a better understanding of the contributions of this paper let us briefly review the state of the art in Internet services characterization and QoS analysis.

1.1 Related work

The analysis of the state of the art in Internet services characterization and QoS reveals that TCP dynamics are subject to extensive study in order to determine their influence on user-perceived quality of service. The behavior of TCP depends of a large number of factors such as the congestion on the path from the client to server [2], the transmission window negotiated at the connection establishment phase [3], the maximum segment size [4], the roundtrip time estimation and the protocol version (Reno, Tahoe, Vegas) [5, 6], that provide different implementations of delayed ACK and window recovery mechanisms. Thus, due to the TCP daunting complexity we note that the current studies focus on specific aspects of TCP performance in restricted simulation scenarios [2, 6, 7, 3, 4, 8, 9], network links on experimental network configurations [5, 10] and selected client to server links [11].

Thus, the analysis presented in the abovementioned studies are constrained by the fact that a real Internet connection is penalized by interfering traffic, which is not captured by simulation models. On the other hand, users navigate the Internet in a random manner. Thus, the quality of service perceived by the user cannot be evaluated by experimental setups or selected client-server paths. It is only through unconstrained measurements of real Internet traffic that an accurate characterization of Internet services and user-perceived QoS can be achieved.

In [12] a real traffic trace is used to explore TCP dynamics in a real network. However, the analysis is restricted to the relation between ACK compression and segment loss, and their influence to TCP dynamics. On the other hand, trace-driven Internet service analysis is performed in [13] and [14]. The random variables describing per session statistics such as asymmetry, number of bytes and packet interarrival times are analyzed empirically but both studies provide no insight on QoS measurements such as transaction latency and use traces that do not include the most popular service in the Internet: the WWW. Regarding the latter service, most WWW studies are based on logs from servers or proxies [15, 16]. In [17] a trace at client side is collected in order to study WWW session characteristics such as duration, size and user behavior aspects. Since all of the abovementioned WWW analysis rely on connection records obtained with logs produced by the client browser or by the server/proxy we note that only application level statistics can be obtained, and, thus, no emphasis is done on user-perceived QoS and how to relate to the underlying network and transport protocol parameters.

More recently, [18] presents a trace of Internet traffic recorded in a wide area link including weekly and daily pattern analysis. However, the analysis is mostly focused on plotting traffic volumes and packet sizes by service. As a conclusion, the lack of studies that evaluate not only the macroscopic characteristics but the perceived quality of service for IP over ATM services motivates the research presented in this paper.

1.2 Network scenario and measurement tool

Our traffic traces are obtained from the network configuration depicted in figure 1. The measurements are performed at the ATM PVC that links Public University of Navarra to the core router of the Spanish academic network (*RedIris* [1]) in Madrid. Rediris topology is a star of Permanent Virtual Circuits (PVCs) which connect the Universities around the country to the central interconnection point in Madrid. From the central RedIris facilities in Madrid a number of international links connect the Spanish ATM academic network to the outside Internet. The measured PVC is terminated at both sides by IP routers. The Peak Cell Rate (PCR) of the circuit is limited to 4 Mbps and the transmission rate in the optical fiber is 155 Mbps.

We note that the scenario under analysis is a representative example of a number of very common network configurations. For example, the most Spanish Internet Service Providers (ISPs) hire ATM PVC links to

[1] http://www.rediris.es

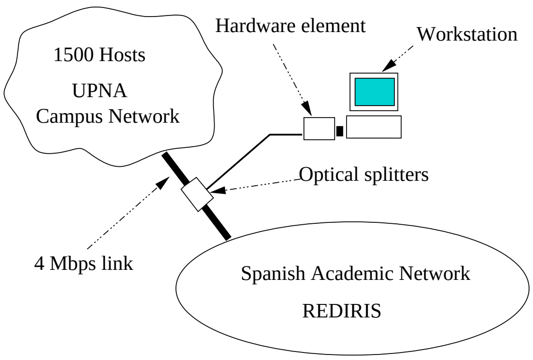

Figure 1: Network measurement scenario

the operator in order to provide customers with access to the Internet. The same situation arises with corporate and academic networks, that are linked to the Internet through such IP over ATM links. On the other hand, our measurements are not constrained by a predetermined set of destinations. On the contrary, they constitute a real example of a very large sample of users accessing random destinations in the Internet. Furthermore, we carefully check the ATM PVC utilization factor and note that it never reaches 50% during the measurement campaign of one week. This sanity check is performed to ensure that the present analysis accurately portrays a general Internet case. Namely, different connections are facing different bottleneck links according to the destination, but the results are not correlated by a potential bottleneck link in the access. Finally, the wealth of data in the trace provides a strong confidence level in the obtained results. Table 1 summarizes the main characteristics of the traffic trace presented in this paper.

Start date	Mon 14/12/98 0:00 GMT
End date	Sun 20/12/98 24:00 GMT
TCP conns	1,700,000
IP pkts	9,000,000

Table 1: Trace characteristics

Our measurement tool is implemented with a dedicated hardware to avoid packet filters effects, that can produce measurement skews as noted in [19]. Measurements are performed on-line instead of collecting a trace at the ATM or IP level and performing off-line analysis. The advantage of this technique is that a large amount of data can be analyzed without interruption since measurements are not so storage-intensive. The dedicated hardware performs extraction of the IP and transport protocol headers, both TCP and UDP. Furthermore, the timestamp resolution for the IP datagram and the ATM cell is 14/12 μs. The offered traffic that can be supported with no cell loss is 300 Mbps. The dedicated hardware relays headers to a UNIX workstation that performs analysis using UNIX threads. One of the major advantages for using threads instead of concurrent processes (namely, processes created with a *fork* system call) is interprocess communication. While the latter may require multiple copies of packet headers in kernel memory the former share the same address space. Therefore, a large throughput between the different threads can be achieved. As a result, a number of threads can be spawned to perform different functions such as TCP connection tracking. Indeed, the measurement tool is able to track the active TCP connections, allowing for a detailed analysis of TCP services, even in high speed environments. We note that a similar hardware/software approach to perform analysis of large volumes of data is adopted in [18]. However, only connection records are provided by the on-line analyzer presented in [18] (OC3MON) whereas our network monitor records the first 100 packets of each TCP connection, thus allowing for a more detailed analysis at the packet level, while keeping the storage requirements at a minimum for long term analysis without interruption. On the other hand, an important advantage of our monitoring tool with respect to OC3MON is that TCP connection teardown is not detected by timeout (64 s.) but the TCP state of every single connection that is captured by the monitor is tracked in real time, according to the TCP segments that are read from the network. Thus, connection establishment and teardown instants are recorded with better accuracy, allowing for a more detailed examination of QoS parameters such as transaction latency and throughput. In the next section we review the macroscopic characteristics of Internet services according to our measurements.

2 Macroscopic analysis of IP over ATM services

The macroscopic analysis of our trace indicates that the ATM link bandwidth is mostly devoted to short TCP connections from WWW accesses. First, table 2 shows the percentage of bytes and packets due to the different transport protocols. Not surprisingly, TCP dominates the sample, since such transport protocol is being used by the most transactional services, like for example the WWW.

Protocol	Percentage of bytes	Percentage of packets
TCP	88.78	79.8
UDP	1.38	5.93
ICMP	0.11	0.37
Other	9.73	13.9

Table 2: Protocols used

Secondly, figure 2 shows the top 10 TCP ports, sorted by number of connections and bytes. Note that some services generate a significant number of connections with very few bytes, such as AUTH(113), LOGIN(49) and DNS over TCP(53). The AUTH service is normally used in conjunction with the FTP, in order to allow anonymous FTP servers to authenticate the client. On the other hand, other services such as Hotline(5501) consume a significant share of network resources (number three in generation of bytes) with very few connections (only 181 in a week!). Hotline [2] integrates multiple services, such as chat, file transfer and news in the same session. As a result, large file transfers produce a significant increase in the number of bytes generated by Hotline connections.

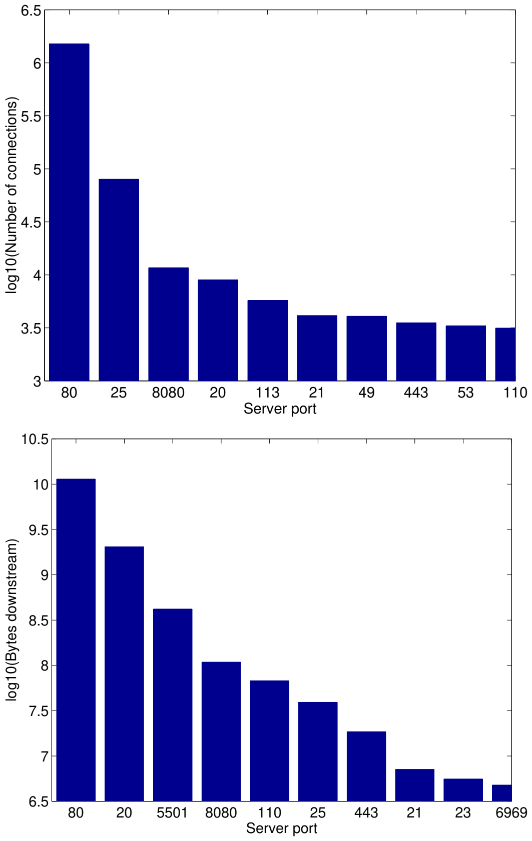

Figure 2: Top 10 ports (log-linear) by number of connections (top) and number of bytes transmitted (bottom)

On the other hand, the traffic trace is dominated by the WWW with 80% of the total traffic in bytes and 90% of connections. We observe from figure 2 that WWW uses port 80 for direct TCP connections and usually port 8080 for proxy WWW connections. The WWW is followed at considerable distance [3] in bytes generation by the FTP (port 20 for data and port 21 for control) and Hotline, which is very similar to FTP due to the file transfers. A small percentage of bytes are due to mail retrieval through POP3 (port 110), mail upload from client to server with SMTP (port 25), virtual terminal services like Telnet (port 23) and secure transactions with HTTPS protocol (port 443). Interestingly, we do not observe any access to the News service (NNTP). Table 3 presents transaction level statistics for the most popular services found in our sample. We note that WWW connections are small in size, with the mean equal to 7.5 KBytes and the 99% percentile equal to 70 KB. We also note a strong asymmetry in bytes transferred from server to client with respect to bytes from client to server, except for SMTP.

Service	Bytes per transaction Cli → Serv	Bytes per transaction Serv → Cli	Duration (seconds)
WWW	551	7552	17.2
SMTP	26394	490	40.0
POP3	69	21494	17.1
FTPdata	0	227603	10.6
Telnet	339	12212	148.4

Table 3: Transaction bytes and duration

Thus, the results of this macroscopic analysis show that the most part of the IP over ATM link traffic is dominated by short TCP connections due to WWW, also noted in recent studies such as [18]. The current Internet scenario differs from the one depicted in previous studies such as [13, 14], in which the traffic trace is dominated by a small percentage of very large bursts due to FTP connections. Nowadays, the load is determined by short WWW connections. On the other hand, user-perceived QoS is determined by the throughput obtained by such connections, that will be studied in the next section.

3 QoS for IP over ATM services

From the previous section we note that the ATM link load and user-perceived QoS is determined by TCP connections produced by WWW accesses, which are presented in figure 3, that shows WWW bytes and connections measured in hourly intervals during the measurement periods of one week. We observe inactivity periods at lunch time and during the night and weekend.

Regarding user-perceived QoS, we note that the QoS metrics commonly adopted for transactional services are transaction latency [8] and throughput, namely bytes transfered from server to client divided by the total duration [4, 20]. We analyze the latter since it provides parsimonious modeling of QoS, while the former depends on a second parameter: the number of

[2] http://www.hotlinesw.com

[3] Note the logarithmic scale in the y axis

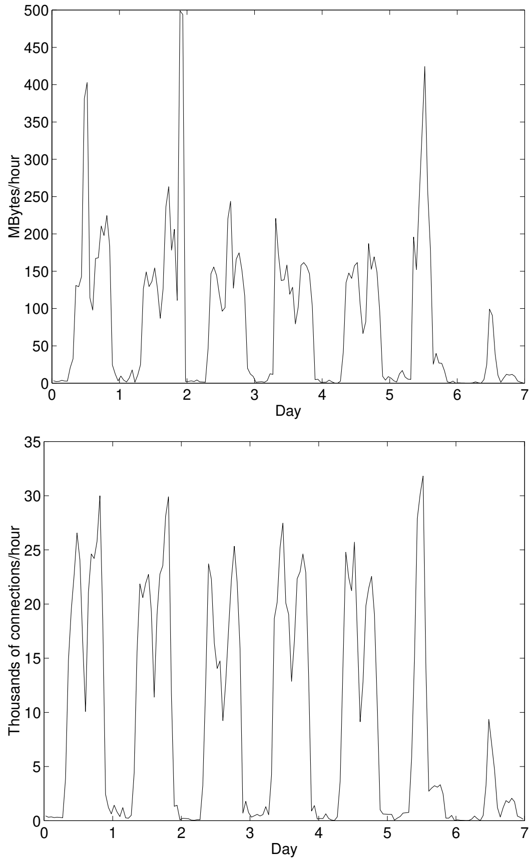

Figure 3: WWW bytes (top) and TCP connections (bottom) from WWW during measurement period

bytes transfered. Regarding the ATM link load, we note that additional parameters are required to characterize the TCP connection load. A common approach is to use the burstiness and throughput [21], which relate to the Maximum Burst Size (MBS) and Sustainable Cell Rate (SCR) as the traffic descriptors at the ATM layer. A commonly used ATM traffic streams model is presented in [21]: Let A_n be the cell arrival process in the time interval n, then the cell arrival process is (σ, ρ)-*constrained* if

$$\sum_{n=k}^{m} A_n \leq \rho(m-k+1)+\sigma \qquad (1)$$

for all k, m such that $k < m$. From equation 1 we note that σ is the burstiness and ρ the throughput of the stream A_n. It can be shown that the arrival process A_n will experience zero loss with a single server with buffer capacity σ and service rate ρ [21]. On the other hand, such (σ, ρ) constrains relate directly to the token rate and token buffer size for a leaky bucket traffic shaper. Furthermore, a multiplex of a number of (σ, ρ)-constrained streams is also $(\sigma\prime, \rho\prime)$-constrained, being $\sigma\prime$ and $\rho\prime$ equal to the sum of the single connection's σ and ρ. Thus, we will use burstiness and throughput as a connection model that captures the QoS provided by the connection and the load to the ATM link.

Since the connection burstiness and throughput will largely depend on the TCP let us briefly describe the dynamics of such transport protocol. A TCP connection begins with a connection establishment phase in order to establish an initial sequence number and to bind the connection to a unique pair of source and destination ports. On the other hand, the transmission buffer size is negotiated between client and server. Following the connection establishment phase the connection enters slow start. The slow-start algorithm makes transmission from the server be "clocked" by ACKs from the client. Each ACK allows for a one segment increase in the transmission window until the slow start threshold is reached. From the initial exponential increase in window size a linear increase follows (each ACK produces a 1/(window size) increase). Slow start makes TCP transmission behave in a stop-and-go manner until window size reaches a value that allows for continuous transmission. Such value is the bandwidth-delay product of the path between client and server, namely the product of the bottleneck link bandwidth and the total roundtrip time (RTT) between client and server. For a throughout description of TCP we refer the reader to [22] and the references therein.

From the above description we note that burst size should increase geometrically in the slow start phase, until the steady-state phase is reached. The connection throughput grows in the same geometric fashion (i. e. 2^i segments per RTT i) until the bottleneck link bandwidth is achieved. Nevertheless, connection throughput depends on the link RTT and packet loss probability [2, 6, 7, 3, 4, 8, 9]. On the other hand, the event that a TCP connection produces a burst of one packet in the first RTT, two packets in the second and so on depends on the RTT, the loss probability and the jitter introduced in the path between client and server, that may separate packets within bursts.

We note that the exact estimation of loss probability and RTT is not feasible. First, loss probability can only be estimated through duplicates of data packets or acknowledgments, but the TCP is not a selective reject protocol and will transmit more packets/ACKs than those actually lost. Secondly, estimates of the RTT based on the time difference between a packet and the corresponding ACK (like the algorithm used by TCP to set the retransmission timer values) suffer several disadvantages: First, the response time at the server side is included in the RTT estimation. Secondly, other TCP features like delayed ACKs make the server send one ACK per several data packets, thus complicating matters for RTT estimation. As a conclusion, we focus our analysis on the characteristics of general TCP connections at the ATM layer, namely burstiness and throughput. We restrict ourselves to a simple estimation of the RTT time scale in order to determine the interpacket gap within bursts.

3.1 Throughput and burstiness characterization

Connection throughput can be derived simply by the ratio between the transfered bytes and the duration. We distinguish between total duration and duration of the data transfer phase. The former is related to the actual QoS perceived by the user, since it takes into account the time elapsed from the user request to the end of the connection. The latter is the actual connection mean offered traffic to the ATM link. On the other hand, the burst size evaluation requires a threshold for the packet interarrival time in order to determine whether subsequent packets belong to the same burst. Such threshold can be derived easily with the peak rate of the ATM link. Since intra-burst transmission takes place at the peak rate we first consider that several packets belong to the same burst if they are transmitted at the peak rate. Interestingly, we find *no bursts of packets belonging to the same connection* at the peak rate in the measurement period.

We note that bursts within TCP connections depend heavily on the path between the server and the client and the TCP dynamics. First, the network induced jitter may separate packets belonging to the same burst. Secondly and most importantly, since TCP connections are short in size a significant portion of the connection lifetime is devoted to slow start, that produces bursts of geometrically increasing size, namely 2^i packets per RTT round i. A typical TCP connection is expected to produce a burst of one packet, followed by an inactivity interval of roughly one RTT, then a two packets burst, with a packet interarrival time that is determined by the bottleneck link bandwidth and the jitter induced by the network. We observe that the packet interarrival time is bounded by the RTT and, thus, it turns out that some estimate of the RTT should be taken into consideration in order to detect such bursts. Specifically, We choose a value of 10% of the RTT estimate as the upper bound for the packet interarrival time within the burst. The burstiness derived with this method is always higher than the burstiness derived with the ATM link peak rate, so it provides a safe upper bound for dimensioning purposes.

The RTT estimation is performed with the initial SYN-SYN handshake. Specifically, we consider the time elapsed from the detection of the SYN from the client to the first segment (ACK to the previous SYN) from the server. We note several advantages of such estimate: first, the server sends an ACK in response to the client SYN with no additional processing at the application layer. The server kernel is in charge of accepting the TCP connection with no intervention of the server application that remains blocked in the *accept* system call. Thus, the impact of server processing time is kept at a minimum. Secondly, the loss of either the SYN segment from client to server or the ACK from server to client produces a retransmission with a three seconds timer [11]. This timer is deterministic because there is no way for the TCP protocol agent neither at the client nor at the server side to achieve an RTT estimate at the connection setup phase. The detection of a retransmission event is thus straightforward, so that anomalous RTT estimation can be filtered out easily. Furthermore, we perform extensive testing of this estimate by using ICMP ECHO packets and conclude that it provides a fair estimate of the RTT when the connection duration is short, namely, the network conditions stationary. Figure 4 shows the histogram of estimated RTTs. Interestingly, we observe two RTT intervals $(5, 100)$ and $(450, 650)$ miliseconds in which the most part of the samples are obtained. Thus, we divide the connection sample into two subsamples according to the estimated RTT intervals, that we denote "low RTT" and "high RTT" in the figures that follow.

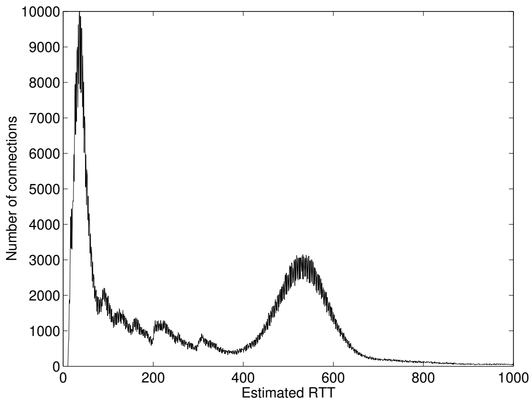

Figure 4: Estimated RTT histogram

Figure 5 shows the connection burstiness according to this criteria and figure 6 shows the throughput. The shaded areas correspond to night and weekend measurement intervals, in which the observed traffic is much lower. Figure 6 shows the throughput in hourly intervals and figure 5 shows the probability of i packets per burst also in hourly intervals, where $i = 1, 2$.

Figure 6 shows that the user-perceived throughput, namely the ratio between the bytes transfered and the total duration decreases significantly in comparison to the data transfer throughput. In order to analyze such performance drop, figure 7 shows the connection establishment phase duration in comparison to the data transfer phase duration. We assume that the TCP connection is in the establishment phase until the first data segment (the HTTP GET) request is issued from client to server. Interestingly, the connection setup time does not include the file lookup time in the server, that would presumably take a significant part of connection duration. Even though such lookup time is not

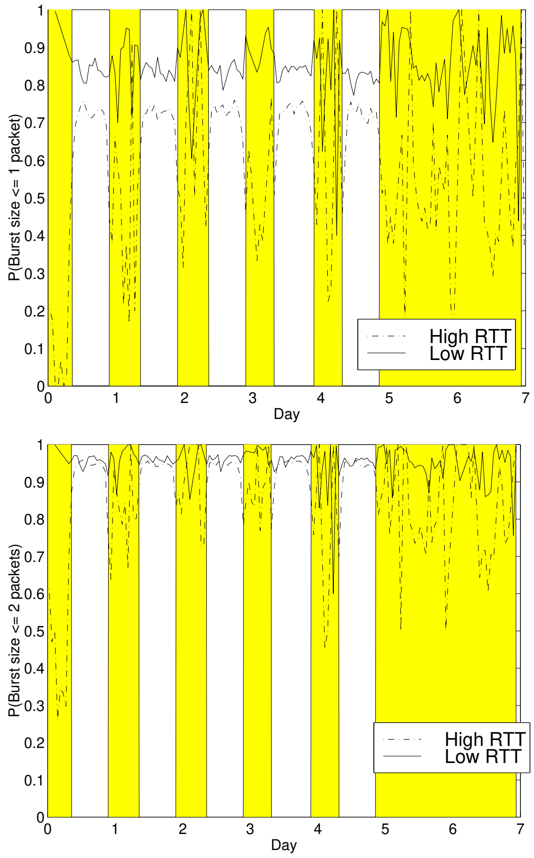

Figure 5: Connection burstiness: $P(burstsize < ipackets)$

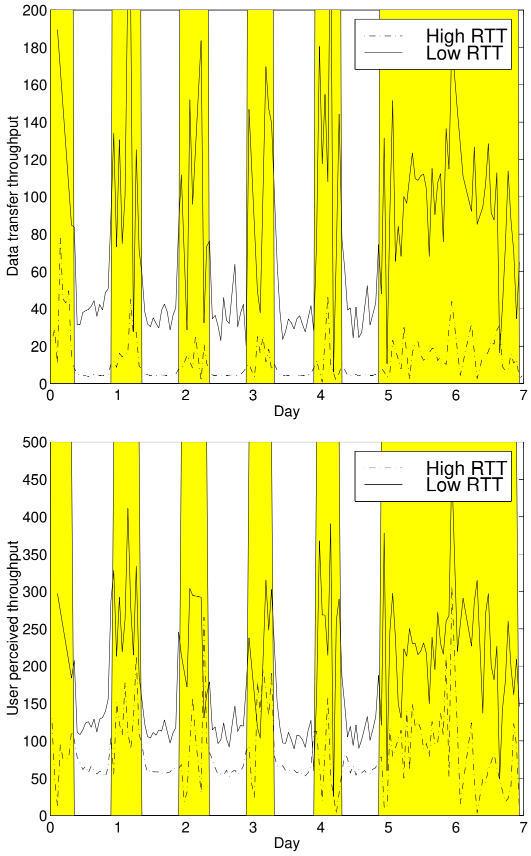

Figure 6: Throughput for total connection (top) and data transfer phase (bottom)

included we note two contributions to connection setup delay: First, the URL submission from the client to the server. Secondly, packet loss in connection establishment phase translates into considerable delay since the TCP retransmission timer takes on high values in the connection establishment phase (in the order of seconds) until more RTT samples are received. Furthermore, the impact of the connection establishment phase is significant since connections are short in duration.

On the other hand, the results in figure 5 show a very low connection burstiness (roughly 95% of bursts contain less than 2 packets) and suggests that a small buffer should be enough to provide zero loss in the link. Indeed, we perform on-line simulations of the ATM link under different shaping conditions and verify this hypothesis. We note that the most TCP connections do not reach the steady state phase and, thus, produce short bursts due to slow start, that are also affected by the jitter and loss induced by the network.

4 Conclusions

An empirical study of Internet services on IP over ATM links is presented in this paper. The ATM link load is primarily determined by short TCP connections, so is the QoS perceived by the end user, which can be measured by the connection throughput. The impact of the connection establishment phase in such user-perceived throughput is striking. On the other hand, the TCP slow start dominates the connection lifetime, since only a few packets (less than 10 with 90% probability) are transmitted per connection. Thus, the influence of the TCP dynamics is critical for user-perceived QoS. Besides this throughput characteristics, we note that connections introduce very low burstiness per connection at the ATM layer.

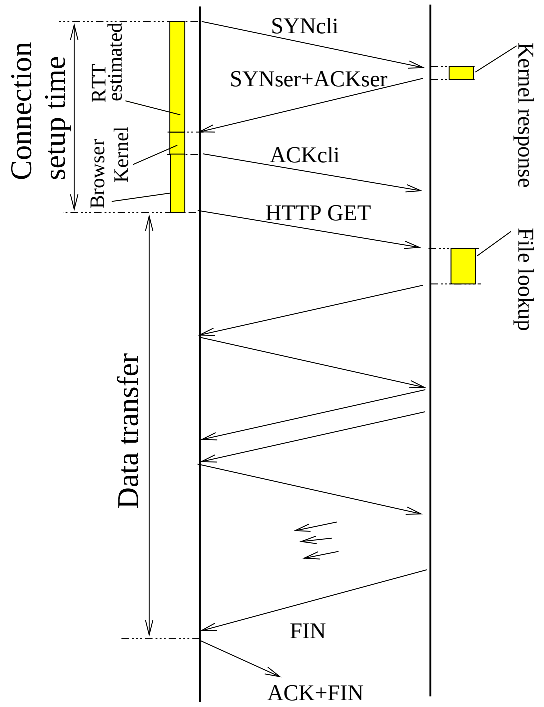

Figure 7: Connection and data transfer phase intervals

References

[1] B. Arnaud. Architectural and engineering issues for building an optical internet. In *Proceedings of SPIE International Symposium on Voice, Video, and Data Communications – All-Optical Networking: Architecture, Control, and Management Issues*, Boston, MA, November 1998.

[2] S. Floyd. Connections with multiple congested gateways in packet switched networks, part 1: One-way traffic. *ACM Computer Communications Review*, 21(5), October 1991.

[3] K. Poduri and K. Nichols. Simulation studies of increased initial TCP window size. RFC 2415, September 1998.

[4] R. Cohen and S. Ramanathan. Tuning TCP for high-performance in hybrid fiber coaxial broadband access networks. *IEEE/ACM Transactions on Networking*, 6(1), February 1998.

[5] K. Fall and S. Floyd. Simulation-based comparison of Tahoe, Reno and SACK TCP. *Computer Communication Review*, 26(3):5–22, July 1996.

[6] J. Mahdavhi M. Mathis, J. Semke. The macroscopic behavior of the TCP congestion avoidance algorithm. *Computer Communication Review*, 27(3):67–82, July 1997.

[7] T. V. Lakshman and U. Madhov. The performance of TCP/IP for networks with high bandwidth-delay products and random loss. *IEEE/ACM Transactions on Networking*, 5(3):336–351, June 1997.

[8] K. Obraczka J. Heidemann and J. Touch. Modeling the performance of HTTP over several transport protocols. *IEEE/ACM Transactions on Networking*, 5(5), October 1997.

[9] A. Romanow and S. Floyd. Dynamics of TCP traffic over ATM networks. *IEEE Journal on Selected Areas In Communications*, 13(4):633–641, May 1995.

[10] A. Kumar. Comparative performance analysis of versions of TCP in a local network with a lossy link. *IEEE/ACM Transactions on Networking*, 6(4), August 1998.

[11] S. Savage N. Cardwell and T. Anderson. Modeling the performance of short TCP connections. Available in http://www.cs.washington.edu/homes/cardwell/quals/quals-paper.ps, October 1998.

[12] J. Mogul. Observing TCP dynamics in real networks. Technical report, Digital, Western Research Laboratory, 1992.

[13] S. Jamin R. Cáceres, P. Danzig and D. Mitzel. Characteristics of wide-area TCP/IP conversations. In *Proceedings of ACM SIGCOMM '91*, 1991.

[14] V. Paxson. Empirically derived analytic models of wide-area TCP connections. *IEEE/ACM Transactions on Networking*, 2(4), August 1994.

[15] S. Glassman. A caching relay for the World Wide Web. In *Proceedings of the First International Conference on the WWW*, 1993.

[16] Marc Abrams, Charles R. Standridge, Ghaleb Abdulla, Stephen Williams, and Edward A. Fox. Caching proxies: Limitations and potentials. In *Proceedings of the Fourth International Conference on the WWW*, 1996.

[17] M. E. Crovella and A. Bestavros. Self-similarity in world wide web traffic: Evidence and possible causes. In *ACM SIGMETRICS Annual Conference on Measurement and Modeling of Computer Systems*, May 1996.

[18] G. Miller K. Thompson and R. Wilder. Wide-area Internet traffic patterns and characteristics. *IEEE Network*, November/December 1997.

[19] V. Paxson. Automated packet trace analysis of TCP implementations. In *Proceedings of ACM SIGCOMM*, September 1997.

[20] D. Estrin R. Cocchi, S. Shenker and L. Zhang. Pricing in computer networks: Motivation, formulation and example. *IEEE/ACM Transactions on Networking*, 1(6):614–627, December 1993.

[21] R. Cruz. A calculus of network delay, Part I: Network elements in isolation. *IEEE Transactions on Information Theory*, 37, 1991.

[22] W. Stevens. *TCP/IP Illustrated, Volume I.* Addison-Wesley, 1995.