

HTTP/TCP connection and flow characteristics

Joachim Charzinski

Information and Communication Networks, Siemens AG, Munich, Germany

Abstract

Currently, most traffic in the Internet backbone and access networks is World Wide Web (WWW) traffic. On the basis of recent long-time traffic traces we present characteristics of WWW traffic for different flow levels, namely port-to-port, host-to-host, and total client access. Using flow duration distributions, we obtain estimates for the point in time when a shortcut connection should be established. Investigations of the correlation structure of different flow characteristics reveal that symmetrical connections occur even at high bit rates. A large number of TCP connections can therefore benefit from symmetric access network bandwidths although HTTP traffic on an average is asymmetric. © 2000 Elsevier Science B.V. All rights reserved.

Keywords: Internet traffic; HTTP; Measurement; Correlation statistics; Shortcut establishment

1. Introduction

Models for Internet traffic are needed both for network dimensioning and architecture refinement purposes. Currently, most traffic in backbone and residential access networks is World Wide Web (WWW) traffic. Although WWW traffic has been studied extensively in the past, some of its characteristics have not been covered before. This paper concentrates on properties of HTTP/TCP connections and flows for single client computers, as they can be observed in the access area. In contrast to other authors, we have carried out high-resolution measurements yielding packet level details over a time period of several weeks.

In Section 2, the traces evaluated here are shortly described and the definitions used for flows are given. The basic HTTP and TCP message sequence is described in Section 3, giving insight into the amount of traffic generated in the different phases of an HTTP connection. General flow characteristics like flow durations, volumes and bit rates are given in Section 4, leading to an observation which brings the distribution of observed flow lifetimes into context with the residual flow lifetimes to be expected. Section 5 focuses on the correlation of different flow characteristics where some basic properties of observed HTTP/TCP traffic are derived, mainly the correlation between flow durations and achieved bit rates and the correlation between upstream and downstream bit rates and traffic volumes.

E-mail address: j.charzinski@ieee.org (J. Charzinski).

2. Description of traces and flow definitions

The results given in the following sections have been obtained from two different measurements, which will be referred to as “Trace A” and “Trace B”. Both traces were collected at local Ethernet segments where *all* the traffic to and from each user’s computer could be observed.

Trace A was collected from May to December 1998, when 100 students’ PCs were connected to their university’s backbone network in Münster, Germany, via ADSL lines. The lines were configured to 2.5 Mbit/s downstream and 384 kbit/s upstream rates, concentrated in two stages onto 15 lines of 10 Mbit/s each and one trunk line of 100 Mbit/s. Usage of the Internet access service was free of charge and there was no dial-up procedure, i.e. computers could have an “always on” mode connection to the Internet. Roughly seven users could be monitored at one time using a modified version of tcpdump [1] on a 10 Mbit/s Ethernet link. According to the packet filter counters, no packets were lost. As all the user groups could not be monitored at the same time due to technical reasons, the link to be monitored was changed cyclically on a weekly basis. In total, during the six months of monitoring, 14 million IP packet headers belonging to HTTP were collected, covering around 480 000 TCP connections.

Trace B was collected during five weeks in March and April 1999, when all traffic at a local Internet service provider (ISP) “Bürgernetz Fünfseenland” close to Munich, Germany, was monitored using the same software as with Trace A. Around 300 mostly residential subscribers shared 30 dial-up lines reaching from low-speed modems to double ISDN lines at 128 kbit/s plus compression. Apart from charges for the local telephone call needed to connect to the ISP, subscribers only paid a yearly flat rate. Here, 43 million IP packet headers belonging to HTTP were collected, covering around 1.6 million TCP connections.

The results given below are mostly those from Trace A, as the range of bit rates is greater in this trace. Trace B was used to check if results could be generalized, and significant differences are mentioned where applicable. Traffic volume and bit rate data include the Ethernet overhead.

For the following evaluations, we used three different characterizations of traffic flows:

- A port-to-port flow (P2P) is equivalent to the traffic in a single TCP connection. TCP synchronize (SYN) and final (FIN) bits were evaluated to determine the beginning and end of a P2P flow.
- A host-to-host flow (H2H) consists of all P2P flows with the same IP addresses. H2H flows were evaluated from the packet header traces using a gap of at least 600 s (10 min) as a flow termination criterion.
- A client flow (CL) summarizes all HTTP traffic to and from a client computer which has no idle period of more than 600 s.

Note that P2P and H2H flows correspond to flow types 1 and 2 as introduced in [2]. As only one service (HTTP) is considered here, H2H flows and C-S flows as introduced in [3] are identical in this context. For a thorough discussion of flows in conjunction with Web traffic, see [4]. The timeout of 600 s was chosen following an investigation of the dependence of the mean short-time bit rate on the duration of the averaging interval, which showed a decrease of the otherwise constant values above around 600 s. The number of flows detected in the two traces in each class are summarized in Table 1.

Table 1
Flows in traces A and B

Trace	P2P	H2H	CL
A	480794	43537	2260
B	1576151	95401	9253

3. HTTP/TCP protocol operation

3.1. Fundamental operation principle

The fundamental HTTP/TCP procedure for downloading a single item from a server to a client is sketched in a packet based message sequence chart in Fig. 1. Comments on non-standard variations of this procedure are summarized in Section 3.2. When a WWW client downloads an item, after resolving the server’s IP address if necessary, a TCP connection is established from the client to the server via TCP’s three-way handshake connection set-up procedure involving two SYN packets and an additional acknowledgement packet [5]. During the TCP data transfer phase, the HTTP protocol [6,7] uses TCP data packets for a “GET” request to the server and for the response back to the client. After the client acknowledges the last data item packet, the server closes the connection, again using a three-way handshake with the FIN bit set in two TCP packets and a final acknowledgement packet.

If the requested data item is bigger than the payload of one maximum length TCP packet (e.g. around 1450 bytes (B) in Ethernet environments), there are more HTTP data packets transmitted and acknowledged before the connection terminates. If persistent or “keep-alive” connections are used, multiple GET requests can share the same TCP connection, leading to a repetition of the part labeled “data transfer” in Fig. 1 without repeating the connection set-up and release overhead.

3.2. HTTP sequence statistics

The percentage values given in Fig. 1 indicate that around 22% of all packets (including “Resets” to abort connections or connection requests) are used to set-up and release TCP connections. Another 36% of all packets are sent upstream from client to server during the date transfer phase, including HTTP GET requests, HTTP options, headers and other methods, and acknowledgement packets for downstream data.

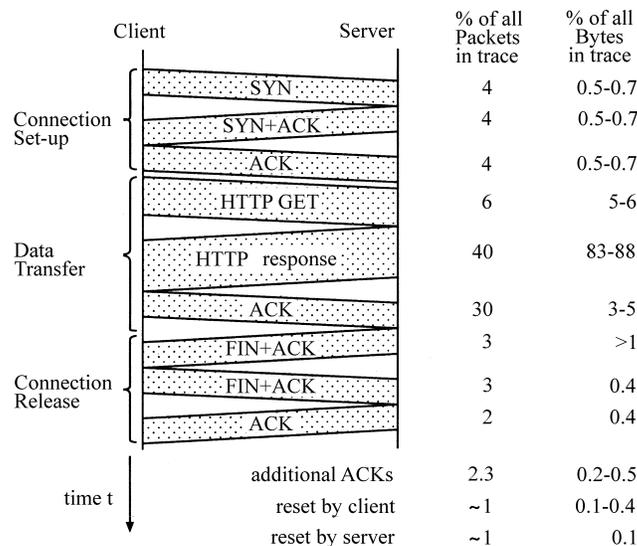


Fig. 1. HTTP/TCP message sequence.

Only 40% of all packets are used for downstream transfers of the requested items. On the other hand, all except the HTTP GET and HTTP data packets are of minimum size, leading to HTTP data packets accounting for between 83 and 88% of all data volume transferred.

These packet count values not only indicate that for an average size HTTP download an one-eighth of the traffic volume is carried upstream, but also that up to one-fourth of all successful TCP connections used for HTTP have eventually been terminated by a connection abort, either by the client or the server, and that not all TCP connections are released properly using the full three-way handshake, which is due to some implementations of TCP showing non-standard behavior.

In special combinations of client and server operating systems and client browser software, systematic misbehavior has been observed leading to the following situation: After a successful transfer, the server sends a FIN packet, which is acknowledged (without the FIN bit set) by the client. Some time later, the client tries to issues another GET request within this half-closed connection, which is answered by a reset packet (RST bit set) from the server, leading to the client repeating the GET request after opening a new HTTP/TCP connection. Other software specific variations on the standard behavior include servers not transmitting the final ACK packet or issuing a separate ACK packet to acknowledge the packet containing the GET request instead of piggy-backing this ACK to the following HTTP data packet.

4. Flow characterization

The distribution of the duration of TCP connections for HTTP has been reported before [8], however, for relatively few samples. From our long-term traces, we were able to extract statistics for port-to-port and host-to-host flows as well as complete client sessions.

4.1. Flow statistics

In Fig. 2, the cumulative complementary distribution function of the flow durations is given for these three flow definitions. A complementary distribution function $C(x)=P[X>x]$ gives the probability to exceed the value given on the abscissa. As both axes are scaled logarithmically, the straight line part in

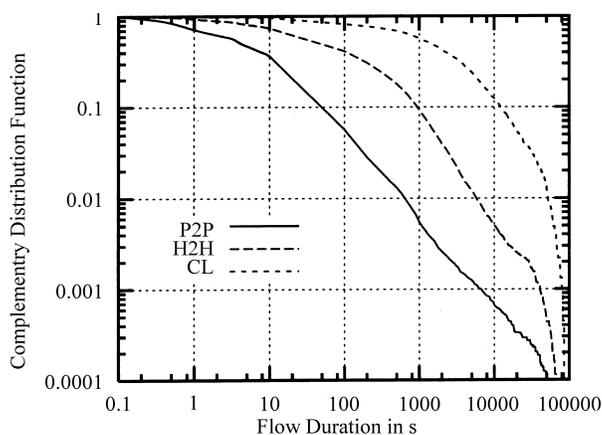


Fig. 2. Distribution of flow durations.

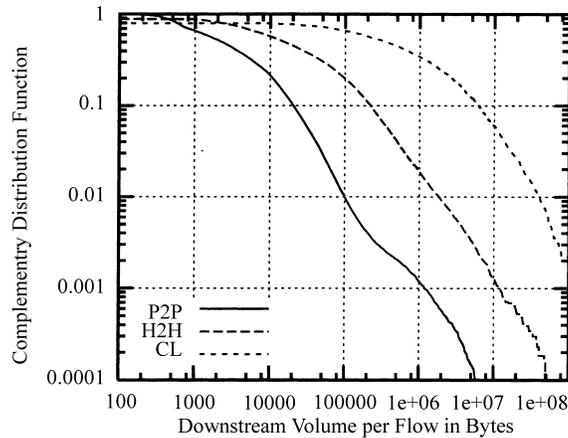


Fig. 3. Distribution of downstream volume.

the distribution of P2P flow durations indicates a power tail distribution, i.e. $C(t) \sim t^{-\alpha}$, for around four orders of magnitude — from 10 s up to one day, which is far beyond the scale of interest in most cases.

The duration of H2H flows is generally longer and the power tail can still be observed over around three orders of magnitude. Complete client Web sessions are again an order of magnitude longer. However, due to the limited number of samples, the statistical basis is too weak to extend the distribution down to probabilities of less than 2%. The results displayed in Fig. 2 have been obtained from Trace A. While only one-third of all port-to-port flows (TCP connections) last for more than 10 s, one-third of all host-to-host flows last for more than 3 min. A third of all client Web sessions is even longer than 1 h. Durations in Trace B show similar effects but (probably due to time based tariffing of the access line phone call) probabilities in the tail are lower.

The distribution of downstream traffic volumes carried by a flow is given in Fig. 3. For HTTP/TCP connections in which a single request is transmitted, the volume distribution is governed by the distribution of response sizes. Response size distributions have been discussed before for academic Web usage [9,10] and have been found to resemble the distribution of file sizes available in the Web [9] and to be similar for residential Web usage [11]. For host-to-host flows, results have been published for an academic backbone network [12].

Comparing port-to-port, host-to-host and client Web access flows in Fig. 3, we find that the duration of downstream flow volumes resembles that of flow durations. Again, the distribution of P2P and H2H flow volumes can be described by power tails, and H2H flows are an order of magnitude larger than P2P flows. However, compared to previous measurements [9,10,12] at least for P2P flows it must be taken into account that 30% of all TCP connections contained more than one HTTP GET request, as indicated by the distribution of the number of GET requests per flow in Fig. 4.

Again, the results given in Fig. 4 have been extracted from Trace A. Results from Trace B are similar but less heavy-tailed. One percent of all TCP connections contained more than 10 GET requests and in 1% of all host-to-host more than 220 items were requested from the same host. These figures reveal that a significant number of clients and servers supported keep-alive or persistent TCP connections for HTTP [7,13]. Most of these connections (more than 96%) were found with HTTP version 1.0 GET requests, indicating that the clients explicitly negotiated keep-alive connections, although this is not the default in

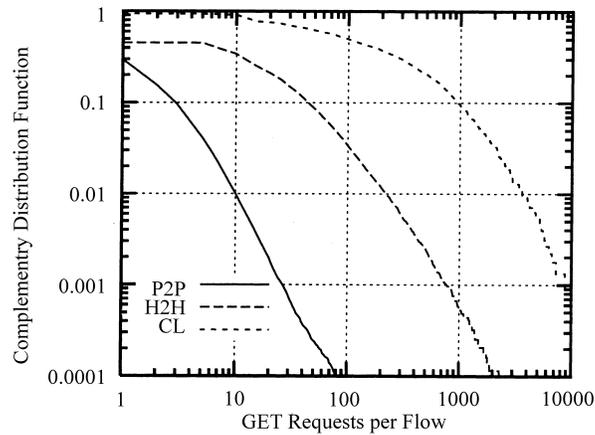


Fig. 4. Distribution of number of GET requests per connection or session.

HTTP 1.0 [13]. All clients observed contacted the Web servers directly, and no proxy connections were found.

For TCP connections with only one GET request, the time to download an item from a server can be expressed in terms of the item size and the mean downstream rate during the connection. The complementary distribution functions of mean downstream rates in P2P, H2H and client flows are depicted in Fig. 5 for Trace A; the corresponding distributions for Trace B are more limited by the access line speed and therefore show a steeper descent above around 20 kbit/s. The curves indicate that:

- even in the existing Internet of 1998, a *mean* downstream bit rate of more than 100 kbit/s could be reached by as many as 4% of all TCP connections by clients connected via high speed access lines;
- the overall shape of the distribution of mean rates is relatively similar for P2P, H2H and client flows;
- host-to-host and client flows contain more idle time than port-to-port flows. This is the cause for the reduction of mean rates seen in Fig. 5. The exact proportion of idle time is a function of both the

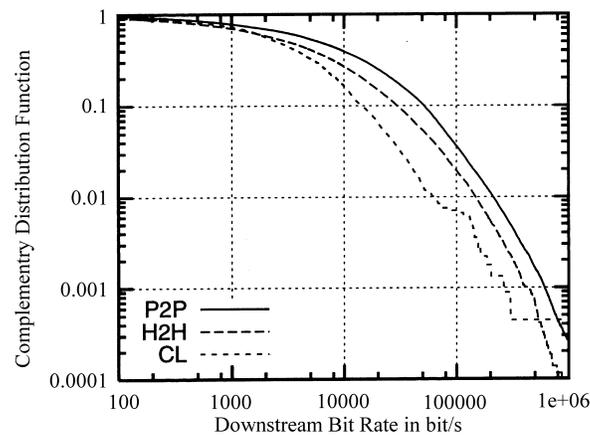


Fig. 5. Distribution of the sustained downstream rates of flows.

Table 2

Summary of locality statistics and mean downstream rates for Trace A. Local traffic is to and from hosts inside the high-speed German research network. The difference between *all* and the sum of *local* and *non-local* is due network addresses for which no associated host name could be resolved via DNS

Locality	P2P flows			H2H flows		
	No. flows	% Flows	Mean rate (kbit/s)	No. flows	% Flows	Mean rate (kbit/s)
Local	57564	12	34.7	2893	7	19.1
Non-local	304509	63	19.6	27419	63	12.6
All	480794	100	20.6	43537	100	12.7

timeout interval used for flow detection and the averaging time for determining the bit rates. However, it still takes significant values even for very short (e.g. 10 s) flow timeouts.

As Trace A was recorded in a network connected to the high-speed German Research Network (DFN WiN), there might be a considerable difference between the rates achieved by connections within this network and connections to outside destinations. In order to achieve a classification of connections, the hostnames involved have been resolved after the measurement and declared *local* if at least one hostname from the same IP network has been identified as part of the DFN WiN. Hosts from IP networks without addresses in the DFN WiN were declared *non-local* and when name resolution was not possible, the corresponding packet records were excluded from the locality evaluation summarized in Table 2. Host names could not be resolved for 25–30% of the communication partners. The mean rates given in Table 2 and the corresponding rate distributions not shown here indicate that the distributions given in Fig. 5 for all traffic are also representative for the non-local traffic up to around 500 kbit/s, whereas local connections could achieve rates that were a factor of 1.5–2 higher than those given in Fig. 5.

As indicated by the distributions in Fig. 4, P2P as well as H2H flows often consist of multiple downloads of data items, which could be a reason for relatively low mean rates and large proportions of idle time in P2P flows. However, a comparison of flows serving just one GET request in contrast to flows with more than one GET request reveals that the corresponding mean downstream bit rates are only 1.1 times as high for flows with exactly one GET request and 0.6 times as high for flows with strictly more than one GET request compared to those given in Fig. 5.

The observation of reduced mean rates in H2H and client flows in Fig. 5 might lead to the conclusion that the current browser software practice of opening several parallel connections between client and server does not succeed in its goal to accelerate downloads. However, this conclusion is only true for the *mean rate* of each flow type. More detailed statistical evaluations averaging on a short time slot basis instead — and thus giving more weight to long living flows — show that the mean instantaneous rates observed in H2H flows are 50–100% higher than those of P2P flows. The mean instantaneous rates of CL flows are even higher, indicating that parallel connections to multiple servers are more effective than parallel connections to the same server.

4.2. Statistics based flow detection

The distribution of flow durations presented in Fig. 2 allows quantification of the distribution of the residual lifetime of a flow if the expired lifetime of a flow is known. Let u denote the expired lifetime of a flow, T_R the residual lifetime and let the cumulative complementary distribution function of flow

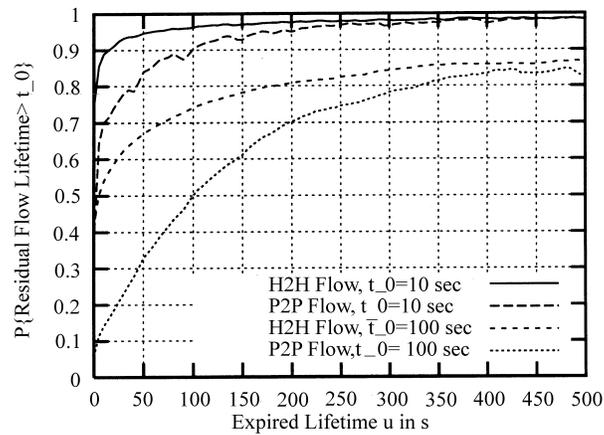


Fig. 6. Probability of a flow exceeding residual lifetime $t_0=10$ or 100 s given it has already lasted for time u .

lifetimes T be denoted by

$$C_T(t) = P[T > t].$$

The conditional probability for the flow lifetime T to exceed $u+t_0$ once the flow has lived for time u is given by

$$P[T > u + t_0 | T > u] = P[T_R > t_0 | T > u] = \frac{C_T(u + t_0)}{C_T(u)}. \quad (1)$$

In a backbone network where shortcuts can be automatically established upon detection of a flow, this equation offers a way to include the time t_0 it takes to set-up such a shortcut in the decision when to establish the shortcut. If one demands that the corresponding flow have a certain probability ε to survive at least this time t_0 after the shortcut is set-up, the decision can be taken at time u after the first packet of the flow such that $P[T > u + t_0 | T > u] > \varepsilon$ according to (1) [14].

Fig. 6 shows the results obtained from a numerical evaluation of (1) using the measured distributions from Fig. 2 and a target residual lifetime of 10 or 100 s for port-to-port and host-to-host flows. Two effects are worth remarking:

- The probability to survive another t_0 increases with the expired lifetime u of a flow. This is a consequence of the “heavy-tailed” distributions of flow durations, implying that the longer a flow has lasted, the larger its expected residual lifetime. Another way to put this is to say “*It is never too late to establish a shortcut for an HTTP flow.*”
- As one would have expected, port-to-port flows need to live longer than host-to-host flows to give the same security for exceeding the same target residual lifetime.

The same technique can also be applied to packet counts or byte counts instead of lifetimes or a combination of two or three of these measures to allow an informed decision as to when to set up a shortcut if a shortcut is only useful after a certain residual lifetime t_0 .

5. Correlation of flow characteristics

The distributions given in Figs. 2–5 are not independent of each other. Therefore they provide only a part of the information needed to efficiently engineer networks that have to carry HTTP traffic, as not all combinations of characteristics appear in HTTP connections with the same probability that could be assumed if the characteristics were independent of each other.

5.1. Bit rate and duration

Fig. 7 indicates the combinations of mean downstream bit rate and connection duration that have been observed for TCP connections (P2P flows) in Trace A with a black dot indicating that a combination has been observed at least once within the 480 000 TCP connections. The gray value is changed towards lighter values on a logarithmic scale indicating that the corresponding combination of mean downstream bit rate and duration of a TCP connection has occurred more often, which can be seen in the central part of the covered area.

When interpreting Fig. 7, it has to be kept in mind that both connection duration and mean downstream rate are plotted on logarithmic scales. If the scales were linear, all points would assemble on the two axes, indicating that connections either have long duration and a low downstream rate or short duration and a high downstream rate. In the logarithmic representation of Fig. 7, however, two additional findings can be stated:

- A minimal TCP connection consists of one client-to-server SYN packet and one server-to-client RST (reset) packet. As packets have been collected on an Ethernet, each of these packets has a size of 60B. In the minimal TCP connection, only the RST packet is in downstream direction, therefore the minimum traffic volume is 480 bits, which corresponds to the limiting line of points in the lower left of Fig. 7. All of these points belong to SYN–RST minimal connections, i.e. this scenario is found through nearly six orders of magnitude of time — from 2 ms up to 1000 s. Whereas for short flows this is the time it takes a server to answer a connection request with an RST packet, the longer flows have more *upstream* volume with the client successively transmitting packets to the server, which is not accounted for in Fig. 7.

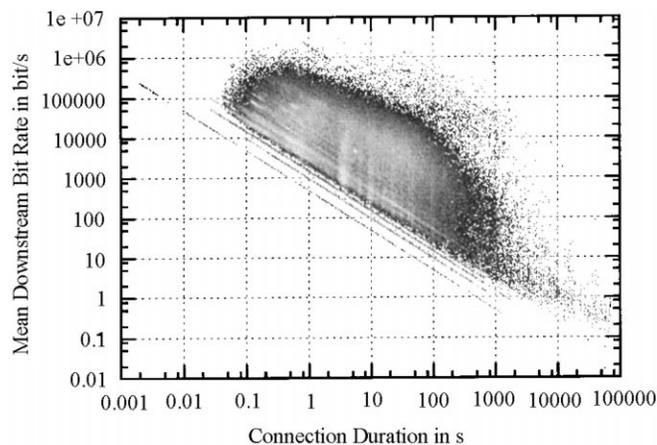


Fig. 7. Correlation of mean downstream bit rate and connection duration.

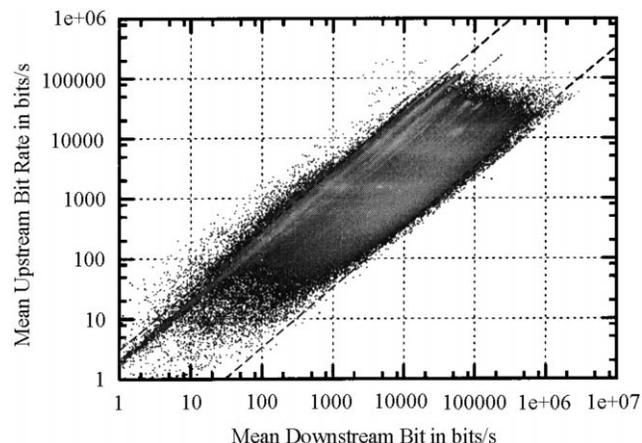


Fig. 8. Correlation of upstream and downstream connection mean bit rates exceed this limit.

- Connection durations and connection bit rates vary over an even wider range of values than the data volume transported in a connection, which corresponds roughly to the distance of a point in Fig. 7 to the line of minimal connections points as explained above and which is well known to have a heavy-tailed distribution [9]. This effect can be confirmed by closer inspection of Figs. 2, 3 and 5. While the ratio between 99th and 1st percentile of downstream connection volumes is only around 200, the same ratio is around 10 000 for connection durations and 20 000 for mean downstream bit rates of connections.

5.2. Upstream/downstream ratios

The same approach is taken in Fig. 8 to correlate the mean upstream and downstream bit rates in a TCP connection. A similar investigation has only been published before by Cáceres et al. [15] comparing upstream and downstream traffic volumes for Telnet and SMTP (simple mail transfer protocol) connections. Fig. 8 shows that over a wide range of scales in the double logarithmic diagram, the ratio of the mean upstream rate to the mean downstream rate of a connection varies only between 3:1 and around 1:30. These ratios are indicated by dashed lines in Fig. 8.

Despite the fact that according to the values in Fig. 1, around 88% of all bytes are transmitted in downstream direction, there are obviously a significant number of TCP connections which have a mean upstream bit rate that is as high as or even higher than the mean downstream bit rate. In order to correctly interpret the maximum values occurring in Fig. 8, we recall that the access line rates for Trace A were limited to 384 kbit/s upstream and 2.5 Mbit/s downstream, so that the mean upstream rates in Fig. 8 could not exceed this limit.

The fundamental properties of Figs. 7 and 8 do not depend on the number of GET requests served in one connection, i.e. the graphs are practically unchanged if only those connections with only one GET request are plotted. Also, the more restricted range of possible bit rates in Trace B is only reflected in a corresponding truncation at the upper rate limits [16].

In order to further explore the ratio of upstream to downstream traffic, the complementary distribution function of this ratio is plotted in a lin-log graph in Fig. 9. As both mean upstream and downstream rates correspond to the same duration of a connection or flow, the ratio of upstream to downstream mean rates

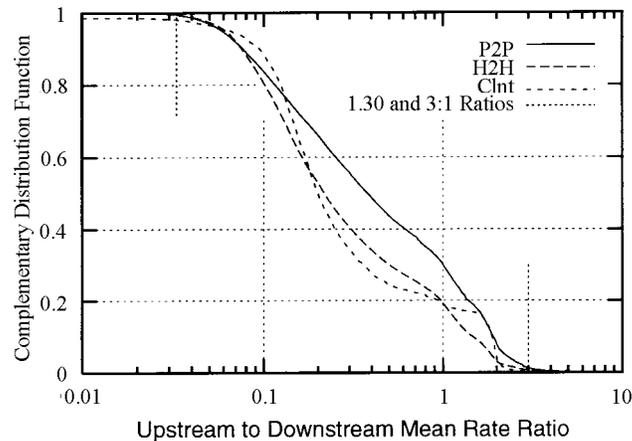


Fig. 9. Complementary distribution function of upstream to downstream rate or volume ratio.

is equivalent to the ratio of the amount of upstream to downstream traffic volume in a flow. The observed limits of 3:1 and 1:30 found above are confirmed by Fig. 9, and they are extended to be valid not only for TCP connections or port-to-port flows — as given in Fig. 8 — but also approximately for host-to-host flows or even complete client Web access sessions. Between 20 and 30% of all HTTP flows observed on the access lines have a higher traffic volume and mean bit rate in upstream than in downstream direction.

The reason behind these ratios can be found in Fig. 10, which depicts the correlation between the downstream data volume in an HTTP/TCP connection and the ratio of upstream to downstream mean bit rates in a log–log plot: There is an evident correlation between the amount of downstream traffic in an HTTP/TCP connection and the ratio of upstream to downstream bit rates. This can be explained by the message sequence in Fig. 1. The parts that vary from one connection to another are the number and size of GET requests and response packets and the number of acknowledgements for response packets. In a connection used to download a very small item, the upstream connection control and acknowledgement packets and the GET request can easily be larger than the sum of sizes of the downstream connection control packets and the response. On the other hand, if a large item is requested, the size of the response will dominate the traffic volume carried in the connection, as is suggested by the numbers given in Fig. 1. The same relation between connection volume and asymmetry has been found for the main data direction in other TCP based applications [17] and for different access line rates [16].

In other words: *The smaller the size of the requested item, the more symmetrical the connection.* This leads to two important conclusions:

- If the access network does not provide for sufficient upstream bandwidth, connections with little response data will experience additional delay. As Fig. 8 indicates, the upstream to downstream ratio is fairly independent of the actual bit rates observed. Therefore, a high upstream bandwidth is necessary when WWW clients have to load pages consisting of many small items when the round-trip delay between client and server is relatively short.
- Even requests for large amounts of data lead to a certain limiting asymmetry of around 1:30. If the upstream bandwidth is too low, HTTP/TCP will not be able to utilize the downstream bandwidth even with large-volume data transfers.

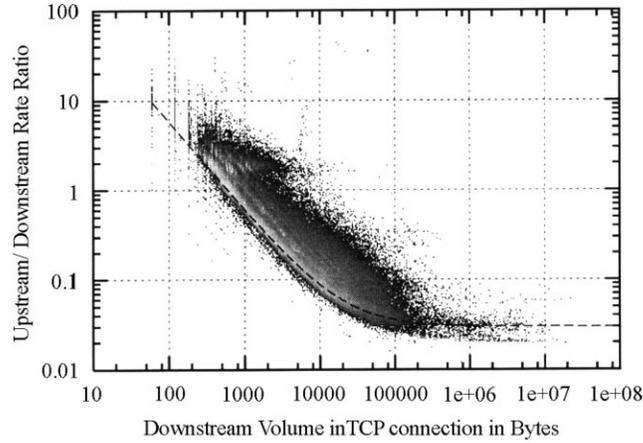


Fig. 10. Correlation of downstream data volume and upstream/downstream rate or volume ratio. Measured values and approximation according to (4).

An explanatory curve has been added to Fig. 10 using a simple mean value formula for the rate ratio $r(v_d)$ as a function of the downstream volume v_d

$$r(v_d) = \frac{v_u}{v_d} \approx \frac{580\text{B} + \alpha n_p \cdot 60\text{B}}{v_d} \quad (2)$$

with the upstream volume being given by the sum of the sizes of three minimum sized packets of 60B each and the mean GET request size of around 400B plus a fraction α of acknowledgement packets per transmitted downstream packet. The downstream volume is roughly

$$v_d \approx 180\text{B} + n_p \cdot 1500\text{B} \quad (3)$$

with n_p counting the full-size downstream data packets. Combining (2) with (3), we get

$$r(v_d) \approx \frac{580\text{B}}{v_d} + \alpha \frac{60\text{B}}{1500\text{B}}. \quad (4)$$

A mean value of $\alpha \approx 0.75$ has been used for Fig. 10, as is suggested by the values given in Fig. 1. The data points above the approximation curve in Fig. 10 are due to larger GET requests, connections with multiple GET requests and connections with a lower maximum transmission unit (MTU). The theoretical minimum of the rate ratio when including TCP, IP and Ethernet overheads as in Fig. 10, is $\frac{1}{50}$ as the delayed acknowledgement option in TCP demands that every second packet be acknowledged by the receiver ($\alpha=0.5$). Therefore, the amount of upstream capacity needed for TCP transfers could be reduced by increasing the MTU or by allowing for more packets to be unacknowledged by delayed acknowledgements.

6. Conclusion

On the basis of long-term high-resolution traffic traces collected in 1998 and 1999, statistical data for HTTP port-to-port and host-to-host flows as well as complete client Web sessions have been evaluated.

The presence of power tails has been confirmed in the distribution of flow durations, flow volumes and number of HTTP GET requests per flow. Mean downstream bit rates also show a sub-exponential distribution tail. It was shown that from the distribution of flow durations, the conditional probability for the residual lifetime of a flow to exceed a certain threshold time t_0 given the age u of the flow can be computed via a simple equation. This approach could assist in informing decisions with respect to the choice of threshold values for establishing shortcuts for host-to-host or network-to-network flows in wide-area networks.

By investigating the correlation between the mean downstream bit rate and the duration as well as the correlation between mean upstream and downstream bit rates of a flow, several effects were described, leading to a fundamental result showing the dependence of the upstream to downstream rate ratio on the downstream volume of an HTTP/TCP connection. Using these results, simple traffic models for HTTP can be improved, and the debate on the symmetry or asymmetry of access bit rates will have to be continued.

Acknowledgements

The author would like to thank Dr. Paul Schlüter, his co-workers at Siemens and his partners at the Universities of Stuttgart and Würzburg, Germany, for their help in evaluating the traces and for the fruitful discussions of the results presented here. Additionally, the anonymous reviewers' comments helped to improve this paper and to clarify the presentation.

References

- [1] S. McCanne, C. Leres, V. Jacobson, Tcpcdump, LBNL Network Research Group. <ftp://ftp.ee.lbl.gov/tcpdump.tar.Z>.
- [2] P. Newman et al., Ipsilon Flow Management Protocol Specification for IPv4, IETF request for comments RFC 1953, 1996. <ftp://ftp.isi.edu/innotes/rfc1953.txt>.
- [3] J.A. Copeland, R. Abler, K.L. Bernhardt, IP flow identification for IP traffic carried over switched networks, *Comput. Networks ISDN Syst.* 31 (1999) 493–504.
- [4] A. Feldman, J. Rexford, R. Cáceres, Efficient policies for carrying Web traffic over flow-switched networks, *IEEE/ACM Trans. Network.* 6 (1998) 673–685.
- [5] J. Postel (Ed.), Transmission Control Protocol, IETF request for comments RFC 793, 1981. <ftp://ftp.isi.edu/in-notes/rfc793.txt>.
- [6] T. Berners-Lee, R. Fielding, H. Frystyk, Hypertext Transfer Protocol — HTTP/1.0, IETF request for comments RFC 1945, 1996. <ftp://ftp.isi.edu/in-notes/rfc1945.txt>.
- [7] R. Fielding, J. Gettys, J. Mogul, H. Frystyk, T. Berners-Lee, Hypertext Transfer Protocol — HTTP/1.1, IETF request for comments RFC 2068, 1997. <ftp://ftp.isi.edu/in-notes/rfc2068.txt>.
- [8] M.F. Arlitt, C. Williamson, A synthetic workload model for internet mosaic traffic, in: *Proceedings of SCSC'95*, Ottawa, Ont., Canada, July 1995.
- [9] M.E. Crovella, A. Bestavros, Self-similarity in World Wide Web traffic: evidence and possible causes, *IEEE/ACM Trans. Network.* 5 (1997) 835–846.
- [10] B. Mah, An empirical model of HTTP network traffic, in: *Proceedings of IEEE Infocom'97*, Kobe, Japan, IEEE Press, New York, 1997.
- [11] J. Charzinski, J. Färber, M. Frank, J. Tölle, The AMUSE residential multimedia trials: phase 1 monitoring results summary, in: *Proceedings of NOC'98*, Manchester, UK, June 1998, pp. 86–93.
- [12] K.C. Claffy, H.-W. Braun, G.C. Polyzos, A parameterizable methodology for internet traffic flow profiling, *IEEE J. Selected Areas Commun.* 13 (1995) 1481–1494.
- [13] B. Krishnamurthy, J.C. Mogul, D.M. Kristol, Key differences between HTTP/1.0 and HTTP/1.1, *Comput. Networks* 31 (1999) 1737–1751.

- [14] J. Charzinski, Good news about heavy tails, in: Proceedings of the IEEE Conference on High Performance Switching and Routing (ATM2000), Heidelberg, Germany, June 2000.
- [15] R. Cáceres, P.B. Danzig, S. Jamin, D.J. Mitzel, Characteristics of wide-area TCP/IP conversations, in: Proceedings of ACM SIGCOMM'91, Zurich, Switzerland, 1991, pp. 101–112.
- [16] N. Vicari, S. Köhler, J. Charzinski, The dependence of internet user traffic on access speed, Research Report No. 246, Computer Science Department, University of Würzburg, Germany, January 2000. <http://www.informatik.uniwuertzburg.de/reports/tr.html>.
- [17] J. Charzinski, Internet client traffic measurement and characterisation results, in: Proceedings of ISSLS 2000, Stockholm, Sweden, June 2000.



Joachim Charzinski received his Dipl.-Ing. and Dr.-Ing. degrees from University of Stuttgart, Germany, in 1991 and 1999, respectively. From 1992 until 1997 he was with the Institute of Communication Networks and Computer Engineering (Prof. Kühn) at University of Stuttgart, working on media access control in ATM access networks and other teletraffic topics. Since 1997, he has been with Siemens AG, Munich, Germany, where he is currently leading a research project on Internet traffic modeling.