Technische Universität Berlin

Forschungsberichte
der Fakultät IV – Elektrotechnik und Informatik

How happy are your flows: an empirical study of packet losses in router buffers

Amir Mehmood*
Nadi Sarrar*
Steve Uhlig**
Anja Feldmann*

* Technische Universität Berlin /
Telekom Innovation Laboratories
** Queen Mary, University of London

Bericht-Nr. 2012 – 07
ISSN 1436-9915

How happy are your flows: an empirical study of packet losses in router buffers

Amir Mehmood
TU-Berlin / Telekom Innovation Laboratories
amir@net.t-labs.tu-berlin.de

Nadi Sarrar
TU-Berlin / Telekom Innovation Laboratories
nadi@net.t-labs.tu-berlin.de

Steve Uhlig
Queen Mary, University of London
steve@eecs.qmul.ac.uk

Anja Feldmann
TU-Berlin / Telekom Innovation Laboratories
anja@net.t-labs.tu-berlin.de

ABSTRACT

Studies of Internet traffic have revealed that traffic is consistent with self-similar scaling, shows long-range dependence, and that flow sizes are consistent with heavy-tailed distributions. However, how such characteristics affect fundamental network properties such as buffer overflows and therefore the loss process and link utilization has not been explored in detail.

Relying on advanced instrumentation via NetFPGA cards, we perform a sensitivity study of the packet loss process within routers for different network load levels, flow size distributions, and buffer sizes.

We find that packet losses do not affect all flows similarly. Depending on the network load and the buffer sizes, some flows either suffer from significantly more drops or significantly less drops than the average loss rate. Very few flows actually observe a loss rate similar to the average loss rate. Therefore, any single flow is very unlikely to observe the global packet loss process. Furthermore, the loss process can exhibit scaling properties.

1. INTRODUCTION

Internet traffic has been shown to be bursty and, in particular, exhibits scaling properties, e.g., [20]. The possible causes include the ON/OFF process of the transmission times [16], the heavy-tailedness of flow size distributions [11,13] , the multi-fractal behavior of TCP [14]. While, we, in this paper, do not focus on scaling itself, we note that scaling appears to be an inherent invariant of realistic traffic. This has been repeatedly confirmed by thorough studies and empirical validations [8, 17].

Rather, we, in this paper, focus on *packet losses*. Note that TCP makes packet losses inevitable. Standard variants of TCP use bandwidth probing to determine the available bandwidth on the network path. This implies that TCP will increase its load on the network until it receives a signal, a lost packet, that the network can no longer support the increased load. Therefore, any simulation or experiments with TCP traffic with none or only very limited packet loss is suspicious since it implies that almost all flows are TCP window limited which is not the case in reality [18]. However, the loss process is not only impacted by the large time scale properties of the traffic process but also the small time scale effects, namely the ones before the knee in the scaling plots which are caused by RTT and TCP effects and are responsible for the multi-scale nature of traffic [13, 14].

So far, most studies of packet losses have focused on path losses [7, 19, 27, 29]. This paper studies the loss process of a single network element. Our experiments show that the loss process exhibits scaling and leads to unfairness between flows. We show empirically that depending on the network conditions, e.g., small or large buffers, low or high congestion, the packet losses are not evenly distributed among the flows of different sizes. On the one hand large flows are positively discriminated. We call such flows *happy flows*[1]. On the other hand small flows are negatively discriminated. We call them *unhappy flows*.

Our methodology relies on tightly controlled experiments where we select specific congestion levels, flow size distributions, buffer sizes, and round-trip-times. Using advanced instrumentation via NetFPGA boards and careful analysis across network layers we can track the loss process and its impact on each individual flow and the flow's TCP congestion window state.

We use this setup to perform a sensitivity study of the effects of network load, flow size distribution, and buffer size on the traffic and note the following key insights:

[1] We use the expression *happy flows* in the same vein as done for *happy packets* in [9].

Flow happiness: The losses observed by individual flows differ across flow sizes as well as within flow sizes, and depend on both the load and the buffer size. Moreover, any single flow is very unlikely to observe the global packet loss process.

Link utilization: Small buffers limit the size of the TCP congestion window, leading to poor link utilization.

Packet loss process: Packet losses are not simply random as assumed by stochastic models of TCP [3] but rather exhibit scaling effects under high load and are highly irregular under low load and large buffers. When buffers are small and the load is low, one can assume that losses are uncorrelated at time-scales below the typical round-trip-time.

The remainder of this paper is structured as follows. Related work is discussed in Section 2. We explain our experimental methodology in Section 3. We perform our global sensitivity study to load, buffer size and flow distributions in Section 4. In Section 5, we refine our sensitivity study on a per-flow basis and study the dynamics of the loss process. We discuss some implications of our work in Section 6.

2. RELATED WORK

In the past many researchers have studied the correlation between packet losses on Internet paths, e.g., [7,19,27]. These approaches typically rely on active measurements for sampling the path properties of the data plane, e.g., send probes every tens of milliseconds. Due to this sampling, the actual loss process has to be inferred from observed losses experienced by the probes. While such an inference process might accurately estimate the average path loss observed by a flow on a given path, understanding how the losses are distributed among the flows that share a router buffer requires to observe the traffic at the buffer. This is the approach that we take in this paper.

Note, previous work typically assumes that each probe samples the loss process independently and is unbiased by: (a) the packet stream which is used for the probing which may be UDP or TCP, as well as (b) the size of the flows used for the measurements. We, in this paper, show that both assumptions are questionable: the choice of the flow size as well as the transport protocols impacts the observability of the loss process.

The literature is not necessarily focused on the loss process itself, but may also consider some of its implications. For example, Sommers et al. [25] focus on measuring loss, delay, and jitter from active measurements to check compliance with Service Level Agreements (SLA). While such measurements are likely appropriate for referring to the overall quality of the data plane as seen by particular flows, they do not refer to the actual loss process inside the router buffers as we do in this paper.

Another area of related work regards the sizing of router buffers, e.g., [4–6,12]. While sizing router buffers is not the focus of this paper, our work sheds light on the consequences of different buffer sizes on the performance of individual flows and thus to application performance. Buffer sizing relies on assumptions about the number of flows that share a link as well as on the traffic burstiness. We do not question those assumptions, but show in this paper that the buffer size limits the size of the TCP congestion window. This in turn affects the throughput achievable by individual flows.

3. METHODOLOGY

To achieve our goal of understanding the packet loss process, we rely on a configurable and flexible testbed that allows tightly controlled experiments. We present in this section our experimental setup and we discuss the design choices for the hardware and software components.

Realistic Traffic Generation: To feed the target buffer with enough traffic, we rely on multiple PCs. We selected Harpoon [24] because of its ability to reproduce flow-level behavior consistent with the Internet traffic characteristics. The two main parameters used for customizing Harpoon are the flow-size distribution and the flow inter-arrival time distribution. Most flows in the Internet rely on closed-loop feedback [22]. Therefore, we use TCP flows for most of the traffic. We also add some UDP flows using a VoIP client [1].

For traffic generation we use 4 Intel Core2 Duo 2.20GHz servers with 2GB of 667MHz DDR2 RAM. Each server has a two dual port Intel 82546 Gigabit Ethernet controllers. We use the 64-bit Linux kernel version 2.6.18 as distributed with Debian 4.0 (*Edgy etch*). Each experimental machine has at least two network interface cards. One is exclusively used for controlling and managing the experiments while the other ones are used for traffic generation. No other traffic was present on the network segments during the experiments. We use the default Ethernet MTU of 1500 bytes.

Harpoon is configured to choose file sizes according to Pareto distributions with $\alpha = \{1.2, 1.5, 2.0\}$ and a mean of $\mu = 110KB$. These choices for the Pareto distribution ensure a finite mean while ensuring that the

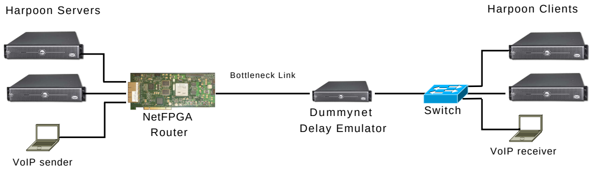


Figure 1: Experimental Setup

generated traffic exhibits variability and scaling behavior. To limit our parameter space, we choose to use an exponential distribution with mean $\mu = 1$ second for the inter-connection times, i.e., the user waiting times between different web requests.

To be able to compare losses seen by TCP with those from UDP, we generate UDP traffic with the open source VoIP framework *PJPROJECT 0.5.10.3* [1]. The generated speech files are using the $G.711$ codec.

Topology Emulation: The network topology we use is the classical *dumbbell* one as shown in Figure 1. All network interfaces are 1 Gigabit Ethernet cards. The configurable network bottleneck is located between the NetFPGA router and the Dummynet delay emulator. Harpoon clients sent Web requests to the Harpoon servers. Using Dummynet [23] we add a delay of 150*ms* to every ACK packet from the Harpoon clients to the Harpoon servers. This delay enables us to emulate round-trip-times which can occur in WAN environments [18]. We explicitly chose to focus on relative large RTTs to better observe the impact of the buffer sizes and the delay imposed by TCP's feedback mechanism. Using the Bro network intrusion detection system [21], we examine the round-trip times within the TCP's three-way handshake and validate that the experimental RTTs have the expected distribution.

Monitoring: Since commercial routers do not provide fine time scale statistics about their buffer occupancy we opt for the NetFPGA as a router. It allows to gather highly accurate buffer statistics. Moreover, we monitor the internal behavior of the TCP stack at Harpoon servers using the *tcphook* [28] Linux kernel module, for two reasons. First, if we want to study the impact of link congestion on the transport layer, we need the ability to monitor TCP's congestion window. Second, the initial value of the slow-start threshold tells us whether TCP is still in its first slow-start phase.

Network Bottleneck: To ensure that the only bottleneck in our setup is the router buffer of the NetFPGA card, see Figure 1, we increase the maximum TCP receive window size to 20MB. This ensures that the transferred file sizes are not receiving window limited [5]. All experiments use TCP New Reno to control the size of the TCP congestion window.

Load	Low	High	Very high
No. of Harpoon sessions	80	200	360
Offered load (%)	50	96	170
Average no. of concurrent TCP flows	140	1250	1700

Table 1: Traffic generation parameters.

Data capture: We capture packet level traces at both the ingress and egress ports of the NetFPGA router. By comparing both traces, we are able to pinpoint missing packets along with transport layer information, e.g., TCP sequence numbers, as well as timing information about when the drop occurred. In addition, we can observe all generated flows from the ingress port trace. Thus we can study the per-flow loss process. We run each experiment for 30 minutes. This duration allows each individual experiment to stabilize. The resulting traces, even though large, can be analyzed within a reasonable time.

Load: To create different network conditions we rely on three different load levels by changing the number of parallel Harpoon sessions on our clients. Note, increasing the offered load can lead to different link utilizations. We distinguish three load levels: *low*, *high*, and *very high*. To determine the necessary number of Harpoon sessions, we run the experiments without link capacity limitations. The lowest load, called *low load*, corresponds to a mean link utilization around 50% which should not impose too much congestion. However, once the load exceeds 50% one can expect degradations in the quality of service, e.g., increased delay and packet loss. Therefore we choose the *high load* scenario in such a way that the resulting utilization will be close to the link capacity. In the *very high load* scenario we intentionally overload the bottleneck link by letting the Harpoon servers generate about 1.7 times the capacity of the bottleneck link. The resulting number of Harpoon sessions[2] and the average number of concurrent TCP flows are shown in Table 3.

Buffer size: To help us choose which buffer sizes to rely on during our experiments, we take into account the recommendations provided by various buffering sizing studies. With our bottleneck capacity of $242 Mbps$ and round-trip time around 150*ms*, the bandwidth delay product (BDP) suggests a buffer size of $3,025$ packets[3]. The scheme proposed by Appenzeller et al. [4] proposes

[2] A Harpoon session is equivalent to flows generated by an Internet user.

[3] The packet size used for the computations in this section is 1500 bytes.

Buffer sizing scheme	BDP	Appenzeller	Tiny buffer
Buffer size (in packets)	3025	86	$20-50$

Table 2: Buffer sizing recommendations for 1250 **flows.**

the following equation to determine the buffer size:

$$B = \frac{RTT \times C}{\sqrt{N}} \qquad (1)$$

where B is the buffer size in bits, RTT is the round-trip-time in seconds, C is the link capacity in bits per second, and N is the number of flows sharing the link. This leads to a buffer size of 86 packets for $N = 1250$ concurrent flows. The "tiny buffer" model [12] recommends buffer sizes around $20-50$ packets. For our experimentation, we mainly use buffer sizes of 256, 128, 64, 32 and 16 packets. We chose the upper bound of 256 packets for the buffer, as it is already 3 times as large as suggested by Appenzeller et al. [4]. During all our experiments, no packet loss occurs outside the bottleneck link.

4. GLOBAL SENSITIVITY STUDY

In this section, we describe the results of a global sensitivity analysis across our parameter space: traffic load, router buffer size, and flow size distribution. We start by studying the impact of different traffic loads and buffer sizes on the bottleneck link utilization (Section 4.1). Then, we show how much the buffer size impacts traffic variability and packet losses (Section 4.2). Finally, we point out the difficulty of sampling heavy-tailed flow sizes within reasonable length experiments for different traffic loads (Section 4.3).

4.1 Link utilization

We start our sensitivity study by examining how an exogenous variable such as link utilization is impacted by endogenous variables such as offered traffic load and buffer size. Note, that the link utilization is a consequence of both of these variables since in particular TCP is always trying to use all resources available in the network: the amount of traffic imposed by the flows that share the bottleneck link as well as amount of buffer available on the path between the traffic sources and sinks. Therefore, link utilization is a result rather than a directly tunable variable.

Figure 2 illustrates the impact of different router buffer sizes on the average link utilization for different offered load levels. Notice the impact of the buffer size. A limited buffer size prevents the traffic from utilizing the available link bandwidth even when many TCP flows are trying to push traffic across the network. Only the relatively large buffer sizes, i.e., 128 or 256 packets enable TCP to fully utilize the link capacity. When the router buffers are relatively small, i.e., 16 or 32 packets, TCP is unable to utilize the link capacity independent of the offered load.

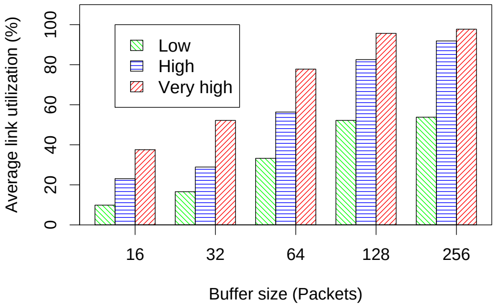


Figure 2: Average link utilization.

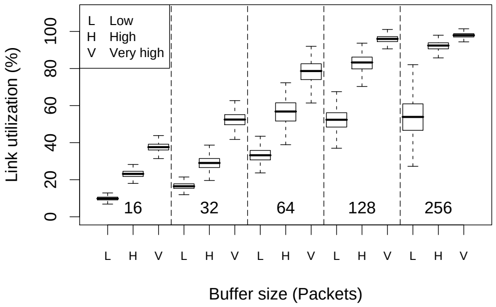


Figure 3: Variability in link utilization.

However, the average link utilization does not tell the full story. We need to also examine the traffic variability. Thus, Figure 3 shows the corresponding standard deviations as well as the quantiles for Figure 2. The standard deviation is a first level summary of the variability and thus gives us a first indication of the impact of buffer size and load on traffic variability. With small buffers, TCP immediately suffers from losses when more than one flow is trying to send a burst of packets. Losses lead to reduced sending rates for each source and smaller overall congestion windows. Hence, the variations around the average link utilization are smaller. When router buffers are larger, each TCP source is able to send larger bursts without incurring losses, leading to higher variability in the traffic. When router buffers are large enough not to interfere too much with TCP's bandwidth probing mechanism, the increasing loads limit the possible vari-

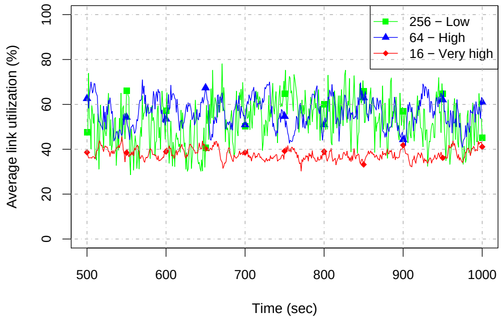


Figure 4: Traffic variability.

ability when link utilization reaches the the link capacity.

Figure 4 compares the link utilization across time for three different buffer size and load combinations at a time granularity of 1s. We chose these three combinations since they exhibit comparable average link utilization but correspond to different offered loads. The combination of a 256 packet buffer and low load exhibits very high variability. The combination of a 64 packet buffer and high load already has significantly less variability. Finally, the combination of a 32 packet buffer and very high load shows even less variability than the other two lines. This illustrates that the average link utilization alone is not sufficient to understand the traffic variability or the bottleneck.

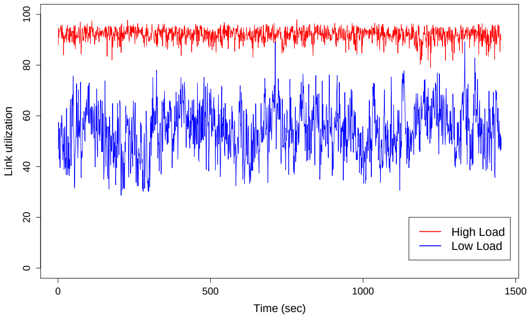
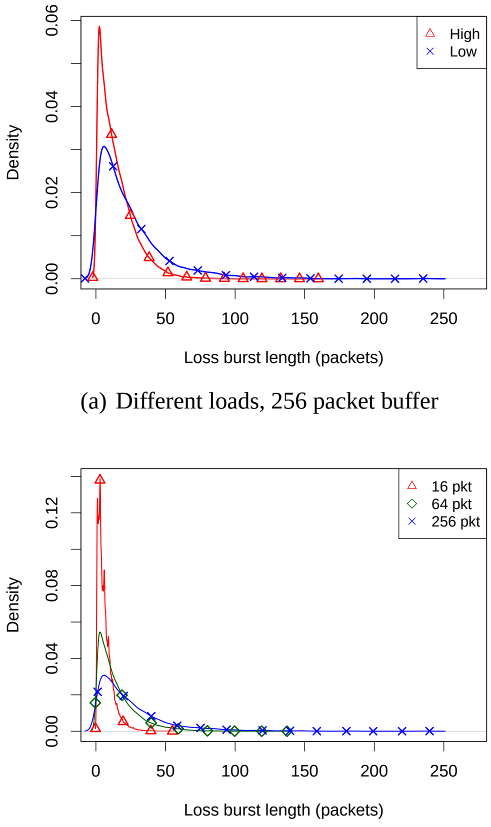


Figure 5: Impact of load on traffic variability.

Finally, we study the impact of offered load on variability. When the offered traffic load is high and router buffers are large, the link utilization is limited by the link capacity and thus traffic variability is also limited. Figure 5 shows the link utilization across time for 1s time bins for two experiments: one with low offered load and one with high offered load, for a buffer of 256 packets. For the high offered load, TCP is not limited by the buffer size. Instead, the link capacity does not allow the TCP senders to increase their rate.

4.2 Burstiness and packet losses

Internet traffic is known to be bursty at several time scales [11, 13, 14, 16]. Here, we are reexamining the traffic burstiness as seen by the router buffer to understand its impact on packet losses. When multiple TCP flows send bursts of packets at the same time, this can result in packet bursts that can exceed the router buffer size, leading to packet losses.



(a) Different loads, 256 packet buffer

(b) Different buffer sizes, low load

Figure 6: Distribution of loss burst length inside the router buffer.

Micro-bursts: When examining the packet loss time series of the NetFPGA buffer we notice that large bursts of lost packets are often separated by a single packet that is successfully sent by the NetFPGA card between two *micro-bursts*[4]. This phenomenon leads to an underestimation of the lost packet burst sizes. Moreover, it

[4] Note, a micro-burst of lost packets can contain packets from a single or multiple parallel TCP flows.

leads to loss burst length distributions that are unexpectedly multi-modal. Therefore, we stitch together *microbursts* of packet losses separated by a single successfully delivered packet. Using more than one packet during the stitching does not significantly change the loss burst length distribution. Under large buffers, e.g., 256 packets, the median micro-burst size is about 20 times smaller than the median of the stitched burst size under high load. Under low load, the ratio is about 36. As shown earlier, with large buffers and under low load, traffic is very bursty, as shown in Figure 3.

The resulting loss burst length distribution is shown in Figure 6(a) for a low and a high offered load scenario and a buffer size of 256 packets. We observe that under low offered load, the tail of the loss burst length distribution is heavier than under high load. This is expected given the higher ratios between the median micro-burst size and the stitched burst size under low load. Under low load, loss bursts are expected to be larger.

The distribution of loss burst lengths is affected by router buffer limitations in the same way as it is by high load. Figure 6(b) shows the loss burst length distribution for different buffer sizes and low load. Small buffer sizes lead to a similar distribution of loss burst length as under high load, with many more smaller loss bursts. Under small buffer sizes, TCP is limited by its congestion window.

4.2.1 *Packet loss*

In principle, traffic burstiness is not a problem. However, since traffic burstiness leads to packet losses it might lead to substantially reduced performance for some flows. However, losses are inevitable with TCP. TCP estimates the available path capacity by generating losses and backing off once it detects a loss. We thus study the average packet loss under different buffer sizes, loads, and flow size distributions.

One of the contributors to Internet traffic variability is the heavy-tailed nature of flow size distributions [11, 13]. We therefore expect to see an impact of the degree of the heavy-tailedness of flow size distributions on the loss process. Figure 7 shows the average loss observed by TCP flows for different buffer sizes, loads, and flow size distributions. Smaller buffers generate high packet losses, larger than 5%, even under low load. This happens because TCP is trying to estimate the available bandwidth on the link based on packet drops, while the drops occur not because of limited bandwidth on the link, but due to too small buffers that cannot handle the packets from the many concurrent TCP flows. When large enough buffers are available, the loss rate reduces dramatically, especially under low load (about 1% loss).

The impact of the heavy-tailedness of the flow size distribution is visible for large buffer sizes (128 and 256 packets). The lower the value of α, the heavier the tail of the flow size distribution, and the higher the packet losses due to a larger number of small flows and the few large flows. When traffic load is high or the buffer size is small the impact of heavy-tails on packet loss is limited by the way TCP is restricted in its burstiness. Note, larger buffer sizes show similar results to those experiments with 256 packet buffers.

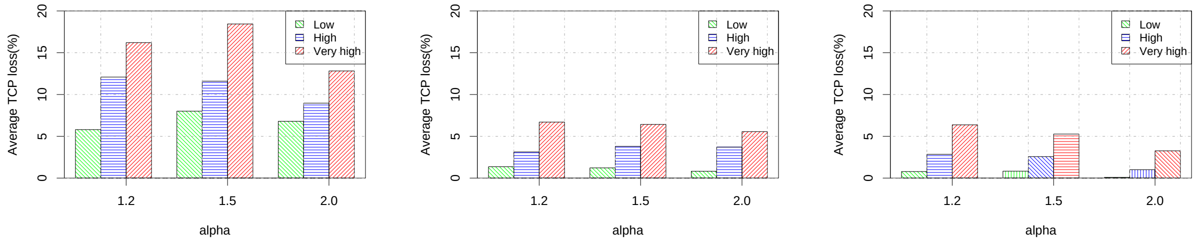
4.2.2 *UDP and sampling the loss process*

Most existing studies of packet loss in the Internet [7, 19, 27] rely on active measurements for sampling the loss process on Internet paths. Such measurements send packets at specific time intervals and infer the loss process based on the observed losses. Sampling the loss process in this way can suffer from two shortcomings: First, one may not sample the periods in which the buffer is full. Second, one may sample the periods in which there is a single free spot available in the buffer. To understand the impact of using such a sampling approach for estimating packet loss we generate UDP traffic using VoIP clients at a rate of about 250 packets per second. This rate is actually higher than used in the literature [7, 19, 27].

Figure 8 shows the average packet loss rate observed by UDP traffic. Note, the loss rate observed by UDP is in general much smaller than the one experienced by the TCP traffic. This is caused by two effects: first the fact that losses are usually occurring in bursts, see Figure 6 and the buffer occupancy process. However, when the router buffer is relatively large (256 packets) and the offered traffic load is low, UDP observes more losses than TCP. This effect is due to an unequal distribution of losses across flows of different sizes, which is examined in Section 5.

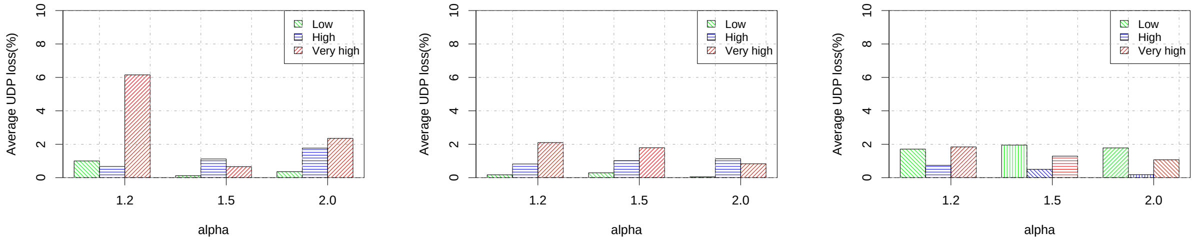
4.2.3 *Buffer occupancy*

The buffer occupancy gives additional evidence and an intuitive explanation for why different loss rates occur for different buffer sizes. For example, for low offered load and relative large buffers (Figure 9), we observe that the mode of the buffer occupancy is relatively small. This implies that for most packets entering the buffer, there is room. However, as the offered load increases, the mode of the buffer occupancy is shifted to the right. Therefore, there each packet entering the buffer will have a non-negligible probability of being dropped. Similar results apply to larger buffer sizes as well.



(a) 32 packet buffer (b) 128 packet buffer (c) 256 packet buffer

Figure 7: Packet loss observed by TCP.



(a) Buffer size 32 (b) Buffer size 128 (c) Buffer size 256

Figure 8: Packet loss observed by UDP.

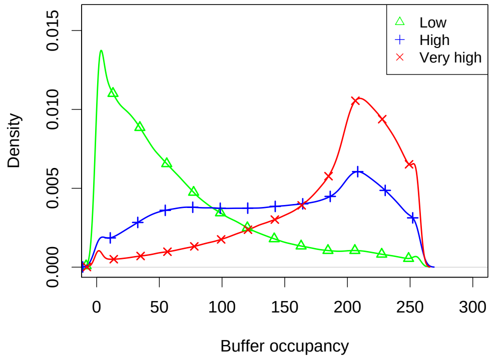


Figure 9: Buffer occupancy for different loads, 256 **packet buffer.**

4.3 Sampling heavy-tails

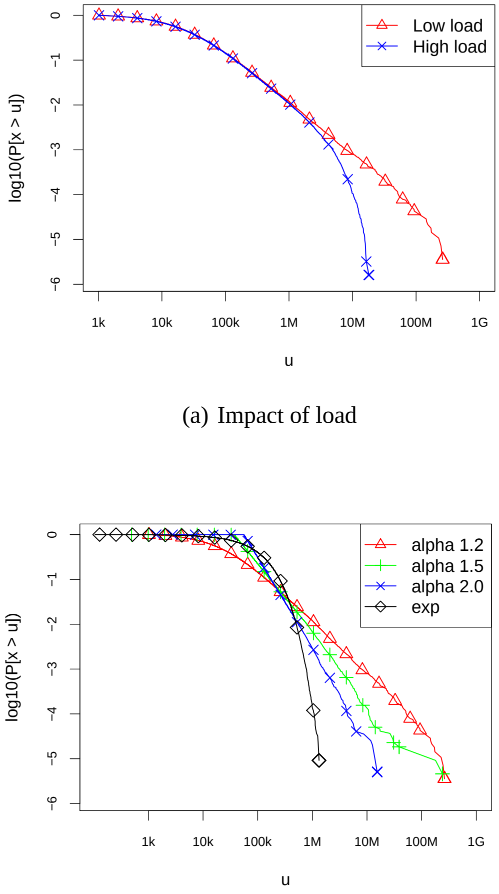
When traffic load is high or buffers are small, traffic variability is limited. Thus, large flows take more time to complete. If the duration of the experiments was infinite, this would not be a problem. In practice however, we have to limit the duration of our experiments. In this paper, we chose to limit our experiments to 30 minutes, which allows us to sample most of the large flows while keeping the duration of the experiments reasonable and the traces manageable.

As mentioned in Section 3, we rely on different flow size distributions for our experiments: exponential and Pareto ($\alpha = 1.2, 1.5, 2$). When using heavy-tailed distributions such as Pareto, some flows are going to be very large. Indeed, some are so large that they may take longer than the duration of the experiment to complete.

Figure 10(a) shows the impact of the offered load on the flow size distributions observed in the traces, for Pareto distributions with $\alpha = 1.2$. We observe that under low load, the tail is nicely sampled. Even flows as large as a few hundreds of MB complete. Under high offered load, flows larger than 10MB hardly complete. Fortunately, our sensitivity study indicates that the impact of the flow distribution on the packet loss process is limited. However, in general one has to pay attention to such sampling issues. Figure 10(b) shows the CCDF of flow sizes for the 4 chosen flow size distributions under low offered load and a buffer size of 256 packets. We observe that the tail is well sampled for all distributions, as expected.

5. FLOW-LEVEL PACKET LOSS

So far, we have treated the loss process as a global phenomenon, i.e., one that takes places across all flows



(a) Impact of load

(b) Distributions under low load

Figure 10: Impact of load on flow size distribution.

that share the buffer and one that does not change over time. However, we already observed in Section 4.2.2 that a limited rate packet flow that samples losses may see a different view of the loss process from one that has a global view of the router buffer. We study in this section the process of packet losses as it applies to each flow individually across different flows sizes and across time.

5.1 Impact of load

We start by studying how different flow sizes are impacted by losses for different offered loads. For each individual flow, the relevant information is not the overall loss rate but the fraction of its packets that have been dropped. Therefore, we compute packet loss rate for each flow as the fraction of packets that were dropped divided by the total number of packets sent by the sender.

Figure 11 shows the per-flow packet drop probability (y-axis, using box-plots[5]) across different TCP flow sizes (x-axis) and for low, high, and very high offered load. Flow sizes are binned into logarithmic sizes. The buffer size is 128 packets and flow sizes are Pareto distributed with $\alpha = 1.2$. Different flow size distributions and larger buffers show similar behaviors and are omitted due to space limitations.

The total packet loss probability in this scenario is rather small with 1%. However, the loss is not distributed evenly across all flows. Under low load (Figure 11(a)), we observe that flows with sizes from 512K to 32M suffer from higher loss rates compared to other flow sizes. The careful reader might remember that in Section 4.2.2 we observed that UDP traffic observes higher loss rates than TCP when the load was low and buffer size large. This is coincidentally the same situation as in Figure 11(a). The UDP flow sizes across the experiments fall in the range of unlucky flows that may actually suffer from higher losses than the total packet loss probability.

When the link utilization is high (Figure 11(b)), a subset of the small flows suffer from larger packet loss probability than the set of larger flows. Note, most of the small flows still have a very small packet loss probability. Only some unlucky flows see more losses than the rest of the flows of a given flow size.

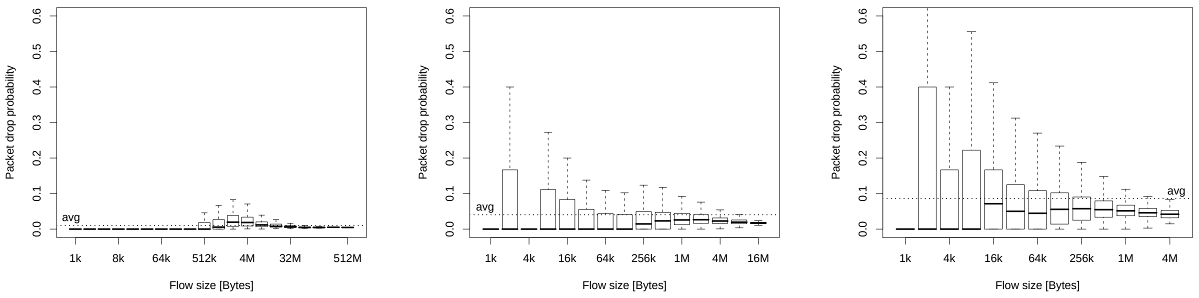
Under high load, the average packet loss probability across all flows sizes increases. Small flows tend to have a few unlucky flows that suffer from very high loss probabilities. For larger flows (larger than 16K) a larger fraction experience packet loss rates of roughly the same rate as the total packet loss rate. Even under very high offered load some happy flows do not observe significant losses.

We thus conclude that whatever the offered load is the observed packet loss probability of a single flow is unlikely to be representative of the total packet loss rate. Even very large flows, which one may expect to better sample the overall loss rate, can observe packet loss probabilities that differ significantly from the overall one. In general, most flows will not observe many packet losses. However, some specific flows might observe unusually high packet loss probabilities—just as some of our UDP flows from Section 4.2.2.

5.2 Impact of buffer size

Next, we examine the impact of buffer size on the packet loss probability for different flow sizes. Section 4 shows that reducing the router buffer size increases packet

[5] Box-plots show the minimum, the percentiles 25, 50, 75, and the maximum.



(a) Low load

(b) High load

(c) Very high load

Figure 11: Per flow-size packet loss probability for different loads (128 **packet buffer).**

loss. We rely on the same flow size distribution as in the previous section (Section 5.1), but instead of varying the offered load, we now vary the buffer size for a low offered load.

5.2.1 Flow happiness

Figure 12 shows the loss distribution across flow sizes for three different buffer sizes: 128, 64, and 32 packets.

With a reasonably large buffer, only specific flow sizes experience unusually large packet loss probabilities, see Figure 12(a). When the router buffer size is small, smaller flow sizes are unhappy, and suffer from high packet loss probabilities. Contrary to scenarios with the high offered load, small buffers do affect all flow sizes consistently.

Even though a high offered load seems to have a similar effect as limited buffer size on packet losses of small flows, load levels and buffer sizes do affect flows rather differently. High offered load creates variability in the way losses are distributed across the different flow sizes—most flows do not see as high a loss rate as the overall loss rate indicates and there are only a small number of very unlucky flows (especially among the small ones) that suffer from unusually high losses. This is the case since high offered load with large buffers does allow some flows to send large bursts. However, these will be dropped when a full buffer is encountered.

In the case of very small buffers, all TCP flow sizes are affected by losses on average, because small buffers cannot absorb large packet bursts. Therefore, a very limited fraction of TCP flows have a chance to send packets without observing losses, no matter how low the load is.

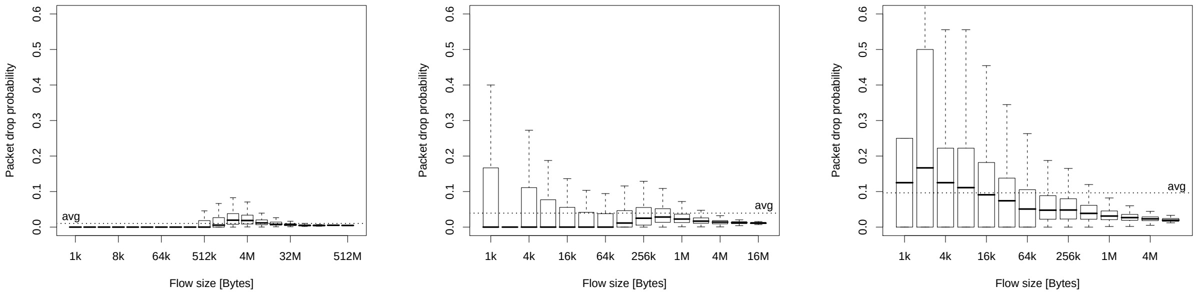
A closer look at the plots in Figures 12 and 11 reveals another difference between high offered load and small buffer sizes. Under high load and large buffer sizes, larger flows tend to observe packet loss probabilities that are closer to the overall average. Under small buffers and low load, it is the small flows that tend to observe packet loss probabilities closer to the overall average. This suggests that under high load, only large flows that last long enough have a chance to properly sample the actual loss probability. When small buffers are the bottleneck, large flows do not representatively sample the actual losses inside the buffer because the buffer limits their TCP congestion window.

Similarly to the high offered load case, reducing the buffer size increases the probability that some small flows will observe high packet loss probabilities. Furthermore, most flows observe much smaller packet loss probabilities than the global one and a limited fraction of the flows observe unusually high packet loss probabilities.

5.2.2 Buffer size and congestion window

Under low utilization and with a large router buffer, we cannot expect that TCP reaches congestion avoidance for small flows. The average congestion window size for each TCP flow, see Figure 13, confirms this intuition. The x-axis of Figure 13 shows the TCP flow sizes while the y-axis shows the distribution of the average TCP window size over the flow lifetime using a box-plot. The top/bottom plot of Figure 13 corresponds to a buffer size of $128/32$ packets. Flow sizes are again Pareto distributed with $\alpha = 1.2$.

Only large flows manage to reach an average window size of the same order of magnitude as the buffer size. Except for very small flows and very large ones, the average window size grows with the flow size until it reaches values in the order of the buffer size. Note, the congestion window can take values as large as twice the buffer size before TCP will be signaled that congestion occurred at the buffer. Interestingly, those very flows for which the TCP congestion window grows beyond



(a) 128 packet buffer

(b) 64 packet buffer

(c) 32 packet buffer

Figure 12: Per flow-size packet loss probability for different buffer sizes (low load).

the buffer size are those who observe the unusually high packet loss.

When the buffer size is small, e.g., 32 packets, as in Figure 13(b), we observe that the average TCP congestion window is limited by the buffer size. Given that the throughout achieved by a TCP flow depends highly on the congestion window, small router buffers limit the performance of TCP even when there is bandwidth available along the path of the flow. In that case, the actual throughput that can be achieved by a TCP flow is much lower than what one might expected from the bandwidth delay product limit, which is 3025 packets in our case. This phenomenon has been observed in residential traffic [18]. We note, that similar observations hold for other flow size distribution including an exponential flow size distributions.

5.3 Time dynamics of packet loss process

Internet traffic has been shown to be bursty, exhibits scaling properties [13,14] and is non-stationary [10,15]. Contrary to what has been assumed in models of TCP [3] about the randomness and stationarity of losses, we would expect that the loss process actually exhibits non-trivial properties over time. Therefore, we examine in this section the loss process across time.

We start with the low offered load scenario and vary the buffer sizes. Figure 14 shows the scaling plots computed on a timeseries of the packet loss process at a time resolution of 1ms. The scaling plot [2] shows, at each time-scale j, the energy contained in the wavelet coefficients ($y(.)$). Since the timeseries has a time resolution of 1ms octave 1 corresponds to a time-scale of 2ms. Each successive octave j offers twice as coarse a resolution as the previous octave. The typical RTT, around 150ms, corresponds to octaves $7-8$. As the loss process might differ across time we use a 3D version of the scaling plot [26]. It shows the evolution across time of the scaling plot computed across over-lapping time intervals. Each time interval over which a single scaling plot takes a 60 seconds time series. We compute a scaling plot every 30 seconds in order to give a smoother look to the 3D plot.

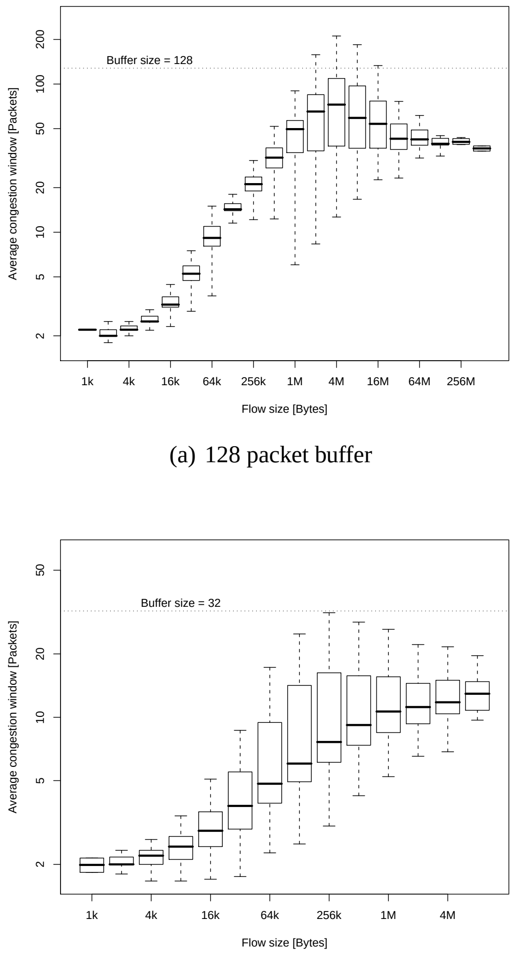
For a large buffer size (Figure 14(a)), e.g., 256 packets, the 3D scaling plot indicates that the loss process exhibits irregularity over time (varying level of consecutive scaling plots), and some possible scaling over time-scales below a typical RTT. When the buffer size is very small on the other hand (Figure 14(b)), e.g., 32 packets, we observe a flat scaling plot for time-scales below the typical RTT, indicating an uncorrelated process. For octaves larger than the typical RTT irregular behavior appears also in this packet loss process.

Increasing the load has a significant effect on the loss process, as can be seen on Figure 15 which again shows a 3D version of the scaling plot. When the offered load is high the loss process exhibits scaling properties at time-scales below the typical RTT.

The difference in the scaling properties of the loss process under small buffers and high load further confirm that two different effects take place. Under high load and large buffers, TCP is allowed to be bursty by the large buffers but has highly variable losses, therefore the scaling. When router buffers are small on the other hand, TCP has not much chance to be bursty, and will therefore generate losses that are uncorrelated.

6. DISCUSSION

Buffer sizing: In principle, our work should be comparable to previous studies that have investigated the implications of buffer sizing schemes [4–6, 12]. Unfortunately, buffer sizing studies make assumptions about the nature of the traffic properties, i.e. TCP senders need to



(a) 128 packet buffer

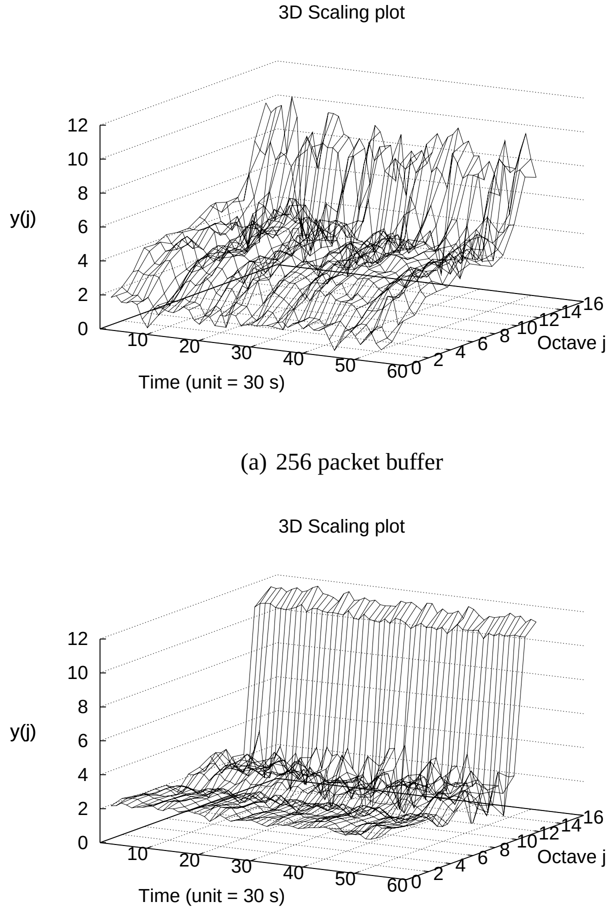
(b) 32 packet buffer

Figure 13: TCP congestion window against flow size (low load).

pace their traffic. These assumptions make sense if the considered traffic is similar to what is observed in very high capacity links where a very large number of flows are multiplexed.

In this paper, we do not make any assumptions about traffic properties, and we have a limited numbers of concurrent flows. Therefore, the burstiness of our traffic is representative of access networks, where a limited number of users are aggregated.

On the one hand, we confirm that relying on router buffers much smaller than the bandwidth-delay product is possible without impacting the ability of TCP to utilize link capacity. On the other hand, given that the TCP congestion window is upper bounded by the buffer size, reducing buffer sizes must be done carefully not to impede on the throughput of TCP.



(a) 256 packet buffer

(b) 32 packet buffer

Figure 14: Impact of buffer size on the packet loss process (low load).

Applications performance: Our study highlights the importance of understanding the flow-level properties of the traffic, e.g., packet loss, under different network conditions, and their consequences on applications performance. For example, an application like Web that usually transfers limited size objects, may suffer from very high losses due to high load, leading to unacceptable Quality of Experience. Bulk data transfer applications that exchange large files might rather be impacted by small router buffers that limit the TCP throughput.

7. SUMMARY

Through controlled experiments, we studied in this paper the relationship between several parameters, including load, router buffer size, and flow size distribu-

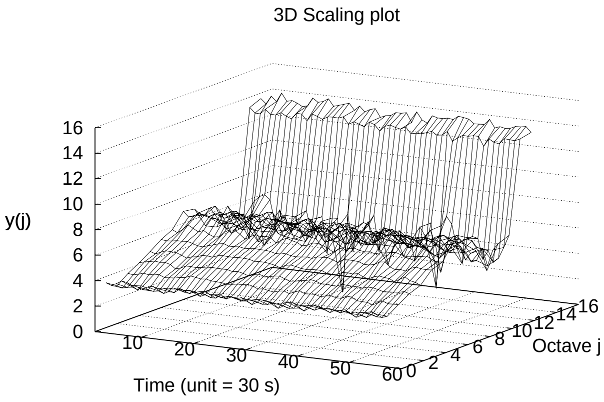


Figure 15: Impact of high load on the packet loss process (256 packet buffer).

tion, on the properties of traffic and the loss process.

Our sensitivity analysis revealed that small buffers have a deep impact on the ability of TCP to use the link capacity. We confirm that both high load and small buffers lead to high packet losses. However, small buffers have a much higher impact on losses than high load.

Surprisingly, we found that packet losses do not affect all flows similarly. Irrespective of the network load and the buffer size, there are few unhappy flows, especially small ones, that observe unusually large losses. On the other hand, most flows, especially large ones, are happy and do not observe high losses compared to the global loss rate. Furthermore, very few flows actually observe a loss rate similar to the average loss rate. Therefore, any single flow is very unlikely to observe the global packet loss process.

Finally, our study of the packet loss process revealed that it can exhibit scaling properties under high load as well as significant irregularities under large buffer sizes. When the buffer size is very small, the loss process is uncorrelated at time-scales below the typical RTT.

In the future, we will study in more details the loss process and its implications on applications performance and quality of experience.

8. REFERENCES

[1] Open Source SIP Stack and Media Stack for Presence, Instant Messaging, and Multimedia Communication. http://ww.pjsip.org/.

[2] P. Abry and D. Veitch. Wavelet analysis of long-range-dependent traffic. *IEEE Transactions on Information Theory*, 44(1):2–15, 1998.

[3] E. Altman, K. Avrachenkov, and C. Barakat. A stochastic model of TCP/IP with stationary random losses. *IEEE/ACM Trans. Netw.*, 13(2):356–369, 2005.

[4] G. Appenzeller, I. Keslassy, and N. McKeown. Sizing router buffers. In *Proc. ACM SIGCOMM*, 2004.

[5] N. Beheshti, Y. Ganjali, M. Ghobadi, N. Mckeown, and G. Salmon. Experimental study of router buffer sizing. In *Proc. ACM IMC*, 2008.

[6] N. Beheshti, Y. Ganjali, M. Ghobadi, N. Mckeown, and G. Salmon. Time-sensitive network experiments. Technical Report TR08-UT-SNL-04-30-00, April 2008.

[7] J.-C. Bolot. End-to-end packet delay and loss behavior in the Internet. In *Proc. ACM SIGCOMM*, 1993.

[8] P. Borgnat, G. Dewaele, K. Fukuda, P. Abry, and K. Cho. Seven years and one day: Sketching the evolution of internet traffic. In *Proc. IEEE INFOCOM*, 2009.

[9] R. Bush, T. Griffin, Z. Mao, E. Purpus, and D. Stutsbach. Happy packets: Some initial results. In *Proc. IEEE INFOCOM*, 2004.

[10] J. Cao, W. Cleveland, D. Lin, and D. Sun. On the nonstationarity of Internet traffic. In *Proc. ACM SIGMETRICS*, 2001.

[11] M. Crovella and A. Bestavros. Self-similarity in World Wide Web traffic: Evidence and possible causes. *IEEE/ACM Trans. Networking*, 5(6):835–846, 1997.

[12] M. Enachescu, A. Goel, T. Roughgarden, Y. Ganjali, and N. Mckeown. Routers with very small buffers. In *Proc. IEEE INFOCOM*, 2006.

[13] A. Feldmann, A. C. Gilbert, W. Willinger, and T. G. Kurtz. The changing nature of network traffic: Scaling phenomena. *ACM CCR*, 28(2), 1998.

[14] H. Jiang and C. Dovrolis. Why is the Internet traffic bursty in short time scales? In *Proc. ACM SIGMETRICS*, 2005.

[15] T. Karagiannis, M. Molle, M. Faloutsos, and A. Broido. A nonstationary Poisson view of Internet traffic. In *Proc. IEEE INFOCOM*, 2004.

[16] W. E. Leland, M. S. Taqqu, W. Willinger, and D. V. Wilson. On the self-similar nature of ethernet traffic. *IEEE/ACM Trans. Networking*, 2, 1994.

[17] P. Loiseau, P. Gonçalves, G. Dewaele, P. Borgnat, P. Abry, and P. Vicat-Blanc Primet. Investigating self-similarity and heavy-tailed distributions on a large scale experimental facility. *IEEE/ACM Trans. Netw.*, 2010.

[18] G. Maier, A. Feldmann, V. Paxson, and M. Allman. On dominant characteristics of residential broadband Internet traffic. In *Proc. ACM IMC*, 2009.

[19] H. Nguyen and M. Roughan. On the correlation of internet packet losses. In *Proc. of ATNAC*, 2008.

[20] Kihong Park and Walter Willinger, editors. *Self-Similar NetworkTraffic and Performance Evaluation*. Wiley-Interscience, 2000.

[21] V. Paxson. Bro: a system for detecting network intruders in real-time. *Computer Networks*, 31, 1999.

[22] R. Prasad and C. Dovrolis. Measuring the congestion responsiveness of Internet traffic. In *Proc. PAM*, 2007.

[23] L. Rizzo. Dummynet: a simple approach to the evaluation of network protocols. *ACM CCR*, 1997.

[24] J. Sommers and P. Barford. Self-configuring network traffic generation. In *Proc. ACM IMC*, 2004.

[25] J. Sommers, P. Barford, N. Duffield, and A. Ron. Accurate and efficient SLA compliance monitoring. 2007.

[26] S. Uhlig. Non-stationarity and high-order scaling in TCP flow arrivals: a methodological analysis. *ACM CCR*, 34(2), 2004.

[27] M. Yajnik, S. Moon, J. Kurose, and D. Towsley. Measurement and modelling of the temporal dependence in packet loss. In *Proc. IEEE INFOCOM*, 1999.

[28] J. Yoo, T. Huhn, and J. Kim. Active capture of wireless traces: overcome the lack in protocol analysis. In *Proc. of WiNTECH*, 2008.

[29] Y. Zhang, L. Breslau, V. Paxson, and S. Shenker. On the Characteristics and Origins of Internet Flow Rates. In *Proc. ACM SIGCOMM*, 2002.