# Computer Sciences Department

A SLA Perspective on the
Router Buffer Sizing Problem

Joel Sommers
Paul Barford
Albert Greenberg
Walter Willinger

UNIVERSITY OF
WISCONSIN
M A D I S O N

# A SLA Perspective on the Router Buffer Sizing Problem

Joel Sommers     Paul Barford      Albert Greenberg    Walter Willinger

Computer Sciences Department         AT&T Labs--Research

University of Wisconsin--Madison

## Abstract

In this paper, we discuss recent work on buffer sizing in the larger context of an ISP's need to offer and guarantee competitive Service Level Agreements (SLAs) to its customers. Since SLAs specify the performance that an ISP guarantees to its customers, they provide critical context for many configuration and provisioning decisions and have specific relevance to buffer sizing. We use a controlled laboratory environment to explore the tradeoffs between buffer size and a set of performance metrics over a range of traffic mixtures for three different router designs. Our empirical study reveals performance profiles that are surprisingly robust to differences in router architecture and traffic mix and suggests a design space within which buffer sizing decisions can be made in practice. We then present a preliminary approach for making buffer sizing decisions within this design space that relates directly to performance and provisioning requirements in SLAs. By comparing our approach with recent and past work on buffer sizing, we show how these prior studies fit into the broader design space we identify and comment on the potential benefits of active queue management in constrained buffer configurations.

## 1 Introduction

Modern routers are extremely complex systems with many features and capabilities intended to improve performance of basic packet switching tasks. At the core of any router architecture is a series of buffers that absorb bursts of packets when the aggregate demand on ingress links exceeds the capacity of an egress link. While the implementation of buffers on router line cards varies significantly from system to system, the objective in their design and configuration is to enable the system, and by extension the network, to meet specified performance targets.

The problem of determining how to configure and size buffers in routers has received significant attention from the research community; see for example the ATM literature of the 1990's, and more recently, the work by Appenzeller *et al.* in [8]. The latter challenges the conventional wisdom that

router buffers should be sized to be (at least) the bandwidth-delay product (BDP) of a link. More precisely, the authors of [8] argue that a buffer size $B$ equal to the product of the capacity $C$ of the link and round trip time $T$ divided by the square root of the number $N$ of long lived TCP flows results in full utilization of the link. The practical significance of this $B = CT/\sqrt{N}$ formula is that it suggests that buffers can be configured significantly smaller than commonly thought, to the point of eliminating the need for up to 99% of output buffers for line speeds of 10 Gbps and above.

The main objectives of this paper are twofold. First, we examine thoroughly the assumptions and premises underlying the startling $B = CT/\sqrt{N}$ result reported in [8]. Second, we provide a novel perspective on the problem of router buffer sizing by relating it to the problem of guaranteeing Service Level Agreements (SLAs). With respect to the first objective, we view the buffer sizing problem within the context of a large design space, which allows for careful evaluation of the tradeoffs that result from different performance metrics, different traffic scenarios, different router architectures and different queuing mechanisms. In particular, we consider performance measures that include delay, loss, goodput, throughput and jitter computed for the aggregate traffic and on a per flow basis. Our traffic scenarios range from homogeneous (*i.e.*, long-lived TCP flows) to highly heterogeneous (*i.e.*, mixture of Web-like TCP traffic and multimedia-type UDP traffic), with a spectrum of realistic aggregate demands. In contrast to all buffer sizing studies to date (a noticeable exception is the empirical evaluation in [8]), instead of relying on idealized router models and abstract queuing simulations, we use 3 different popular commodity router setups (two Cisco and one Juniper) and configure them for different sizes of both drop tail and RED queues. We believe this provides a highly realistic perspective on how packets will be handled in routers in the live Internet. By systematically exploring this large design space, we show that throughput is relatively insensitive to differences in router architecture, buffer size, and traffic mix—an indication that other performance measures might be more useful for making informed buffer sizing decisions. Other performance metrics by and large tend to be

much more sensitive. These findings enable us to broadly assess prior results for sizing buffers. We do this by mapping four sizing methods to specific points in our design space, and comparing and contrasting their impact on performance. Our results clearly demonstrate the performance risks of the $B = CT/\sqrt{N}$ method, and what performance can be gained with the somewhat larger buffers that result from other methods. We also observe that RED queues provide improvements in both aggregate and per flow performance profiles in restricted buffer configurations.

As far as the second objective is concerned, the key observation is that our broad experimental approach enables us to consider how buffer sizing and configuration relate to meeting the performance objectives spelled out in detail in SLAs. Since SLAs specify the performance that an ISP guarantees its customers, they provide critical context for many configuration and provisioning decisions. In particular, we argue in this paper that SLAs should be a primary consideration in buffer sizing decisions since networks are managed to provide a level of service specified by these guarantees, *i.e.,* this is what network providers and customers care about the most. In other words, we argue that buffer sizing decisions must explicitly account for the tussle space defined by ISP economics, router hardware design, and network performance measures [11].

By experimenting with a set of "toy" SLAs that resemble real-world SLAs, we fix a set of canonical performance requirements and illustrate how the proposed perspective exposes new factors that contribute to a more informed decision making process for router buffer sizing, especially at the edge of the network. While fixing performance requirements using SLAs adds a new dimension to the buffer sizing problem, it is apparent from our empirical study that this new perspective creates intriguing new problems (*i.e.,* engineering for robust SLA compliance, quantifying the risk of SLA non-compliance, measuring SLA (non-)compliance) and that an analytic treatment of this question poses formidable technical challenges that are beyond the scope of this paper.

The potential implications of the $B = CT/\sqrt{N}$ result have motivated other researchers to scrutinize the work in [8]. In particular, Dhamdhere and Dovrolis [12, 13] address several of the key open issues and highlight the fact that the traffic model and performance objective used in making buffer sizing decisions are critical. Of the two, the former is perhaps the most difficult to address due to the well known variability and complexity of packet traffic in the Internet [22]. Determining the value $N$ specified as the number of "long lived TCP flows" (*i.e.,* flows that exit slow start) for a given link is challenging, and so is estimating the value $T$ for any non-trivial topology. Dhamdhere and Dovrolis also demonstrate effectively the impact that open-loop versus closed-loop TCP transfers for a given buffer size can have on performance. A distinguishing feature of our work is that we advocate a broad experimental approach. We argue

that a "one size fits all" model for traffic or measure of performance is highly unlikely to expose the impact of realistic traffic behavior or relevant performance measures on buffer sizing decisions. Therefore, the issues of traffic models and performance metrics must be broadly considered, and our study addresses them by taking a comprehensive and flexible experimental approach.

The rest of this paper is organized as follows. In Section 2 we discuss related research efforts on the problem of router buffer sizing. We provide the details of our experimental environment and the configurations used in each of our tests in Section 3. In Section 4 we discuss the results of our experiments that highlight the tradeoffs between performance and buffer size for different test and traffic configurations. Our SLA-based analysis is described in Section 5. In Section 6, we discuss several issues not directly addressed in our study and conclude.

## 2 Related Work

Villamizar and Song are commonly credited for establishing the conventional wisdom for router buffer sizing decisions [30]. Their work was based on measurement of an RS/6000-based implementation of a router, up to 8 simultaneous long lived TCP flows, varying buffer sizes, and both drop tail and RED queues. They found that a buffer size of $CT$ guarantees full utilization of the outgoing link. Morris expanded this work in a simulation study by examining the behavior of a large number of long lived TCP flows competing for a buffer sized at $CT$ [24, 25]. He reported performance degradation for both the end host, and in terms of fairness and utilization. Several additional studies have considered the effects of router buffer size through queuing analysis and in simulation (see, *e.g.,* [9, 18, 26]). A common feature among these and other queuing theory-based studies has been a lack of accounting for the closed-loop nature of TCP traffic flows. Of particular relevance in many of the more recent simulation-based studies has been the consideration of Web-like traffic workloads instead of only long-lived TCP flows. The importance of considering workload models that account for both the feedback regulation inherent in TCP and the heterogeneity of Web-like traffic was demonstrated in [12] (for related earlier work, see Joo *et al.* [20]).

The recent interest in the problem of sizing router buffers has been spurred by the Stanford study of Appenzeller *et al.* [8]. In addition to the articles mentioned in Section 1, a series of recent papers have been concerned with additional details of sizing buffers, controlling synchronization in routers with tiny buffers, and fairness [15, 27, 31, 32]. The primary focus of these papers is on core routers with a high degree of statistical multiplexing, and arguments are made for the feasibility of extremely small buffers (*i.e.,* tens of packets) under certain conditions. The problems of un-

derstanding the behavior of different traffic mixes and active queue management (AQM) schemes are identified as open issues which we address directly in this paper.

The question of the "right" performance metric for buffer sizing decisions has also attracted some attention. Appenzeller *et al.* focus in [8] on *link utilization*, which has been shown in [12] to be oblivious to significant loss rates that would likely be unacceptable to network operators. Dukkipati and McKeown [14] argue that *flow completion time* (FCT) is the most important metric since it best reflects what users care about most. The difficulty with FCT as a performance metric for buffer sizing is that as an end-to-end metric, it is beyond the purview of any single network within which specific buffer sizing decisions have to be made. Gorinsky *et al.* suggest a formulation of the buffer sizing problem that considers end-to-end goodput at the right performance target (directly related to FCT), and show through a series of simulations using different versions of TCP that small buffers can still result in high goodput [19].

The related problem of understanding performance in the context of AQM, specifically Random Early Detection (RED) [17], has been widely examined [16]. Many of these studies have focused on RED tuning and performance evaluation, *e.g.*, [10, 21] which consider the context of Web performance in particular. The RED study by May *et al.* bears some similarity to ours in its use of a commercial router and different traffic mixes [23]. It considers performance with four different buffer sizes and shows that throughput is relatively insensitive to the choice of RED parameters. Interestingly, the results also show that throughput is relatively insensitive to buffer size, but that observation is not highlighted in the paper.

As far as providing an SLA perspective of the buffer sizing problem is concerned, we are not aware of any treatment of the details of service level agreements in the research literature. This is largely due to the fact that SLAs are considered proprietary by ISPs. A portion of the tutorial by Shaikh and Greenberg addresses SLAs at a high level [28]. There is also some general information available from ISPs like Sprint [6], AT&T [1], or NTT [5] that outlines the general performance guarantees for their backbones.

## 3 Test Setup and Traffic Scenarios

Our laboratory testbed, shown in Figure 1, was organized as a dumbbell topology. At its core was a bottleneck OC3 (155 Mb/s nominal) or Gigabit Ethernet (GE) link connected between either two Cisco GSR 12008 routers, or between a Juniper M320 and a Cisco GSR 12008 (routers A and B in the figure). Both OC3 and GE bottlenecks were used with the two-GSR configuration, and only the OC3 bottleneck was used with the M320-GSR configuration. Some detailed differences between these architectures are discussed below. In general, the primary flow of traffic was in the direction of router A to router B (left to right in the figure). Synchronized Endace DAG cards were connected via optical splitters to the links on either side of router A (either Cisco GSR or Juniper M320) for the purpose of comprehensive packet-level measurement. DAG 4.3 cards were used for GE and a DAG 3.8 card was used for OC3[1].

At each end of the topology were 14 hosts (28 hosts total) running Harpoon [29] to generate a variety of traffic scenarios. Traffic from these workstations was aggregated via two Cisco 6500 routers to and from the bottleneck link. These hosts ran either Linux 2.6 or FreeBSD 5.4 and were equipped with either one or two Intel Pentium 4 processors, at least 1 GB RAM, and an Intel Pro/1000 network interface cards. Each host also had another network interface for management traffic, which flowed over a separate physical network (not shown in Figure 1). The TCP stacks on each host were configured with 64 KB receive windows and were configured to be SACK-capable.

Linux hosts running NetPath [7] were interposed in the testbed to perform propagation delay emulation. Two round-trip time distributions were used. The first distribution ("intra-continental") was uniformly distributed between 20 and 80 milliseconds with a mean of 50 milliseconds. The second distribution ("inter-continental") was uniformly distributed between 140 and 260 milliseconds with a mean of 200 milliseconds. We monitored these systems during experiments and periodically reconfigured our DAG cards to run calibration tests on the NetPath systems to ensure that load was well-distributed and that they were able to handle maximum offered loads.

### 3.1 Router Architectures

The three router configurations used in our tests each have significantly different capabilities with respect to the specific line/interface card attached to the bottleneck link, the amount of memory available for packet buffers, and how particular features are implemented (*e.g.*, in software or specialized ASICs). Even between the two Cisco GSR line cards, there are many significant differences [2]. While there are certainly other architectural differences, notably between the Cisco GSR and Juniper M320 chassis, our focus here is on the packet forwarding path specifically as it relates to a single (potentially) congested egress interface.

**Cisco GSR / OC3** This is a low-end line card (termed "engine 0") with features (*e.g.*, RED) primarily implemented in software running on a general purpose processor (MIPS R5000). The entire board contains a total of 128 MB DRAM packet memory (64 MB available for transmit, 64 MB for receive), shared across four

---

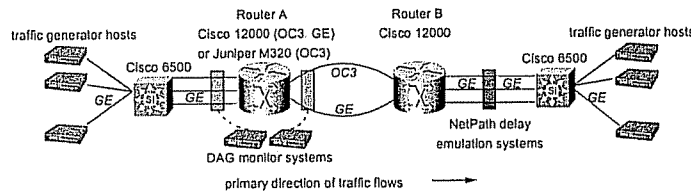[1]The DAG software version used was 2.5.5.

3

Figure 1: Laboratory testbed. Multiple Gigabit Ethernet (GE) links connected Cisco 6500 aggregation routers to two routers separated by a bottleneck link of either OC3 or GE. Router A was either a Cisco GSR 12008 or a Juniper M320. Synchronized Endace DAG monitor cards captured traffic on either side of this bottleneck router. Linux hosts running NetPath were interposed in the testbed to perform propagation delay emulation.

physical interfaces. In addition to DRAM, there is a 128 KB fixed FIFO transmit buffer on each interface.

**Cisco GSR / GE** This is a more sophisticated line card (termed "engine 3") with RED and other features implemented in special-purpose ASICs. There are four physical ports on this line card and a total of 512 MB DRAM packet memory available (256 MB transmit, 256 MB receive). In addition to DRAM, there is a 512 KB fixed FIFO transmit buffer on each interface.

**Juniper M320 / OC3** This physical interface card (PIC) housed in a flexible PIC concentrator (FPC) contains four physical OC3 ports. As with the GSR, packets are buffered both on the ingress FPC/line card and the egress FPC/line card [4], and, like the GSR engine 3 line card, RED is implemented in a hardware ASIC. It is important to note that JUNOS imposes a hard upper limit of 50 milliseconds on buffers configured for a given interface [3][2].

Finally, there are implementation differences in how each line card above handles the division of local buffer space among multiple physical interfaces. Since our focus is on a single congested egress interface, we leave the problem of understanding the impact of multiple, simultaneously congested interfaces on a single line card for future work.

## 3.2 Traffic Scenarios

We used four traffic mix scenarios in our tests and a range of offered loads with each mix. They were as follows:

**Long-lived TCP sources** This setup consisted solely of long-lived TCP flows. Offered load was varied by changing the number of flows. At minimum, we used enough flows to be able to saturate the bottleneck link. The maximum load was an overload scenario where each flow had effectively one packet worth of bandwidth available per round-trip time. For example, with the OC3 bottleneck and 50 millisecond mean RTT setup, we used between 30 and 1200 sources.

**Web-like TCP sources** In this setup, files drawn from a heavy-tailed distribution were transferred across the testbed to create self-similar bursty traffic. We used average offered loads of 60%, 90%, 100%, and 110% of the bottleneck capacity. For the web-like TCP sources as well as the long-lived TCP sources, we configured 90% of the flows to have maximum segment sizes of 1448 bytes, and 10% of the flows to use 512 bytes. File transfers were initiated using a fixed population of on/off threads over the duration of an experiment which results in a closed loop system.

**Web-like TCP with low-rate UDP sources** In this scenario, we used self-similar web-like traffic at an offered load of 50%, and created UDP constant-bit rate (CBR) flows of 64 Kb/s to produce overall offered loads of 60%, 90%, 100%, and 110% of the bottleneck capacity. In addition, we experimented with a fixed ratio between web-like traffic and UDP CBR traffic of 90%/10%, respectively, and tuned overall offered loads to be 60%, 90%, 100%, and 110% of the bottleneck capacity.

**Web-like TCP with high-rate UDP sources** In this scenario, we again used self-similar web-like traffic at an offered load of 50%, and created UDP CBR flows of 1 Mb/s to produce overall offered loads of 60%, 90%, 100%, and 110% of the bottleneck capacity.

There were two configurations of traffic direction. In the first, all data traffic flowed in the direction of router A to router B (left to right) in Figure 1. In this setup, the reverse (right to left) direction consists entirely of TCP ACK traffic. In the second configuration, data traffic flowed in both directions in equal proportions, on average, according to the desired offered load.

## 3.3 Buffer Configurations

The key additional dimension in our tests was to experiment over a broad set of queue configurations. For each of the traffic mixes and offered loads, traffic directions, and round-trip time distributions, we ran experiments using both drop-tail and RED queues, over a range of sizes. We set the output queues on the bottleneck interfaces on both routers A and B

---

[2]This limit was confirmed with Juniper technical support. For core-class routers like the M320 and T640, a relatively low upper limit is set on the maximum buffer size. For edge routers like the M20, there are higher limits.

identically for each experiment No other buffer or queue in our testbed was modified from its default setting.

Table 1 shows drop-tail queue lengths in numbers of packets used for three bottleneck and mean round-trip time configurations. Our settings follow a quasi-logarithmic distribution, which we created by starting with the bandwidth-delay product and dividing by successive factors of four and multiplying by one factor of four. Output buffer sizes on Juniper routers are configured in terms of microseconds. To arrive at a buffer size in microseconds, we multiplied the number of packets by the time taken to transmit a 1500 byte packet at OC3 or GE speed. Note that we explicitly selected a queue size of 1 as a special case to investigate.

Table 1: Quasi-logarithmic drop-tail queue settings (in number of packets, assuming 1500 byte packets) Bandwidth-delay product appears in boldface

| Bottleneck | Mean RTT | | Queue Sizes | | | | |
|---|---|---|---|---|---|---|---|
| OC3 | 50 msec | | 1 | 39 | 156 | 624 | 2496 |
| OC3 | 200 msec | | 1 | 156 | 624 | 2496 | 9984 |
| GE | 50 msec | 1 | 65 | 261 | 1044 | 4167 | 16668 |

Table 2 shows the RED configurations used in our 50 millisecond round-trip time experiments. These configurations were drawn from Cisco's default recommendations. By convention, we used the Cisco recommended setting (appropriately translated) for the OC3 Juniper tests. The small, medium and large settings indicate the three different RED queue thresholds used in these experiments. While there is a large literature on RED tuning, experiments with additional configurations were beyond the scope of our study.

Table 2: RED configuration settings for 50 millisecond round-trip time tests (in number of packets assuming 1500 byte packets)

| | small | | medium | | large | |
|---|---|---|---|---|---|---|
| | $min_{th}$ | $max_{th}$ | $min_{th}$ | $max_{th}$ | $min_{th}$ | $max_{th}$ |
| OC3 | 50 | 150 | 225 | 675 | 400 | 1200 |
| GE | 333 | 1000 | 1500 | 4500 | 2667 | 8000 |

## 3.4 Testing and Analysis Protocol

Tests using long-lived TCP sources were run for 3 minutes and all other tests were run for 10 minutes. Between each experiment, the DAG traces were moved to a separate host for offline analysis.

For each experiment, we calculated aggregate and perflow throughput, goodput, loss, delay, and delay variation (jitter). We first processed the DAG traces taken on either side of router A to extract delay and drop information[3]. We then threw away the first and last 30 seconds of the traces before calculating aggregate and per-flow statistics.

---

[3]We also compared the drop information with router interface counters as a sanity check

In total, we ran approximately 1200 experiments which produced about 1 terabyte of *compressed* raw DAG packet header data.

# 4 Buffer Sizing: Sensitivity Properties

Performance measures that are robust or insensitive to a wide range of networking-related uncertainties are appealing for service providers and customers alike. For service providers, they offer the hope of coping with many of the unknowns associated with user-generated traffic demands and operating the network in an economically sound manner. From the customer perspective, the existence of robust performance measures avoids the need to specify a detailed traffic profile or application mix and still obtain acceptable service from the network.

In the following, we discuss the robustness of performance metrics from the customer perspective, while Section 5 looks at the same issue from a provider's point of view. The design space within which we explore sensitivity issues related to the buffer sizing problem accounts for the different traffic scenarios and different router architectures and buffer configurations detailed in Section 3, as well as for a variety of different performance metrics. Because of limited space, we discuss in the following throughput, delay, and loss, but not goodput and jitter. To illustrate our main empirical findings, we show results of only a small subset of the full set of experiments that we ran for all possible combinations of performance metric, traffic scenario, offered load, router architecture and buffer configuration. In particular, we do not show results of experiments with bidirectional traffic and 200 millisecond average round-trip times, however they are consistent with the results shown below. The selected plots are meant to be representative for the discussions at hand, but may differ in detail from comparable plots. Our focus here is on qualitative comparisons and less on quantifying particular differences.

## 4.1 Performance Profiles: Aggregate Traffic Statistics

We first consider the case where router A can be viewed as an access router (Cisco GSR/OC3 with drop-tail queue) that is fed by traffic generated by long-lived TCP sources. In Figure 2 (top row), we show the performance profiles that result from running this setup for 5x6=30 different buffer size/traffic load combinations. A separate curve is plotted for each source configuration. The buffer size is on the x-axis and the metric of interest is on the y-axis. The three plots show average throughput (left), delay (middle) and loss (right).

To gauge the variability of these average-based performance profiles, we depict in the bottom row in Figure 2 the sets of 30 cumulative distribution functions (CDFs) associated with these profiles. For example, the plot on the left shows the 30 throughput CDFs resulting from running this setup for the 30 pairs of buffer size and traffic load combinations, with specific corner cases labelled. The CDF plots in the middle (delay) and on the right (loss) are similarly constructed. A common feature of these CDF plots is that with the exception of some of the corner cases, they are tightly concentrated around their means (i.e., they are almost step functions). This implies that almost identical performance profiles would result when using percentiles rather than averages; that is, the percentile-based profiles would deviate only minimally from the average-based ones shown in the top row.

Upon closer examination, Figure 2 reveals a surprising degree of insensitivity of the throughput performance profile (top left) to a wide range of changes in buffer size and offered load. In contrast, the delay profile (top middle) shows the expected increase in delay with larger buffers, while the loss profile (top right) reflects the common wisdom that losses generally increase with higher traffic loads. To demonstrate that these features are by and large insensitive to the underlying traffic model and/or router architecture, we show the performance profiles resulting from running the same experiments, but with the long-lived TCP sources replaced by the heterogeneous traffic scenario consisting of Web-like TCP sources in Figure 3, and with the Cisco GSR/OC3 router replaced by a Juniper M320/OC3 system, also fed by Web-like TCP sources in Figure 4. Plots for the two remaining traffic scenarios and with a Cisco GSR/GE system in place for router A are not shown but have similar characteristics. Together, Figures 2–4 confirm and provide additional support for the concerns expressed in [13] regarding the exclusive reliance on the throughput metric in [8] when advocating the $B = CT/\sqrt{N}$ result. In fact, the performance profiles make it clear why throughput is not a very useful metric for buffer sizing, except from perhaps a router hardware design standpoint, and that other metrics such as loss and/or delay are better for making a more informed decision.

To put some of the previously proposed buffer sizing techniques into perspective, Table 3 shows buffer sizes in numbers of packets derived from four different proposed formulas. In particular, Table 3 shows the values corresponding to (i) the traditional bandwidth-delay product (BDP), (ii) the $B = CT/\sqrt{N}$ formula advocated in [8] (Stanford), (iii) the BSCL (buffer sizing for congested Internet links) scheme proposed by Dhamdhere et al. [12], and (iv) Morris's flow-proportional queuing method (FPQ) [24]. To derive these values, we calculated the number of flows long enough to have exited slow start and used this value to parameterize the $CT/\sqrt{N}$ and BSCL models. We used the total number of flows for the FPQ model. Additionally, we followed the

Table 3: Comparison of buffer sizing schemes for infinite source and web-like traffic setups for OC3 bottleneck and 50 millisecond mean round-trip time. Values indicate buffer size in numbers of packets for the traditional bandwidth-delay product (BDP), the $B = CT/\sqrt{N}$ formula advocated in [8] (Stanford), the BSCL (buffer sizing for congested Internet links) scheme proposed by Dhamdhere et al. [12], and Morris's flow-proportional queuing method (FPQ) [24].

| Infinite source load (flows) | BDP | Stanford | BSCL | FPQ |
|---|---|---|---|---|
| 30 | 624 | 113 | 121 | 180 |
| 60 | 624 | 80 | 59 | 360 |
| 150 | 624 | 50 | 780 | 900 |
| 300 | 624 | 36 | 2085 | 1800 |
| 600 | 624 | 25 | 4695 | 3600 |
| 1200 | 624 | 18 | 9915 | 7200 |
| Web-like traffic offered load | BDP | Stanford | BSCL | FPQ |
| 60% | 624 | 84 | 414 | 2767 |
| 90% | 624 | 68 | 651 | 4140 |
| 100% | 624 | 65 | 725 | 4565 |
| 110% | 624 | 60 | 872 | 4958 |

methodology in [12] to empirically derive other parameters required for BSCL. In essence, the values of Table 3 provide indices into the performance profiles of Figures 2–4.

From the perspective of the throughput performance profiles, all 4 methods would do well in achieving high throughput and, judging from the shape of the profiles, so would many other methods. However, some differences become clear when considering the delay and loss performance profiles. For one, considering the buffer sizes proposed by the Stanford model along with the loss performance profiles of Figures 2–4, there is a clear tradeoff between choosing small buffers and risking poor performance in terms of high loss rates. Moreover, using the BSCL formula tends to yield much larger buffers which, considering the delay performance profiles, is at the cost of incurring significant delay. Likewise, the FPQ scheme has a tendency to keep losses low even when it comes at the expense of unreasonably large delays. In short, Figures 2–4 demonstrate that buffer sizing could benefit from a new perspective that provides critical context for configuration and provisioning decisions in general. We will put forward such a perspective in Section 5.

## 4.2 Performance Profiles: Per-Flow Traffic Statistics

In addition to "what" performance metric(s) to consider for the buffer sizing problem, there is also the issue of "how" the metric(s) in question should be computed. In Section 4.1, the metrics were computed based on the aggregate customer traffic. In the following, we present our empirical findings for the same three performance metrics, but now computed on a per flow basis. Using the same set of experiments as in Figures 2–4, we computed in each case throughput, delay, and loss for each individual flow encountered during the experiment, and plot in Figures 5–7 the resulting averages to obtain the throughput, delay, and loss performance profiles.
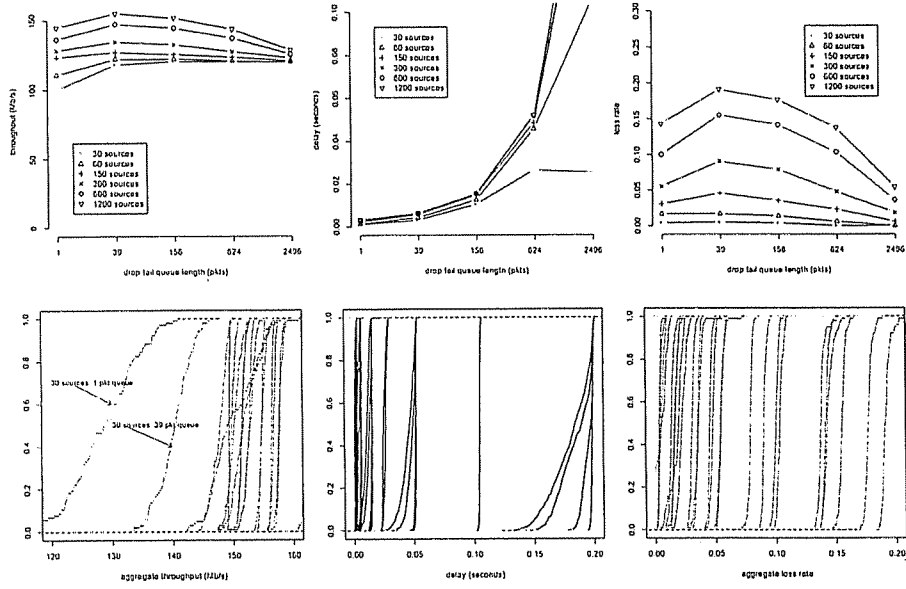
Figure 2: Aggregate results for Cisco OC3 with infinite sources and drop-tail queuing discipline. Profiles of mean throughput, delay, and loss shown on top row and corresponding CDFs shown along bottom row.
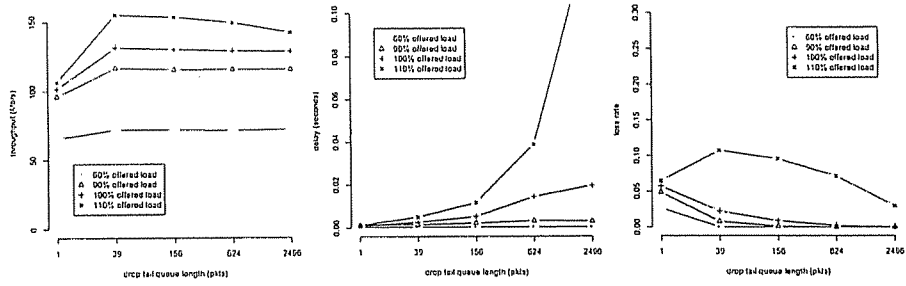


Figure 3: Aggregate profiles of mean throughput, delay, and loss for Cisco OC3 with web-like sources and drop-tail queuing discipline.
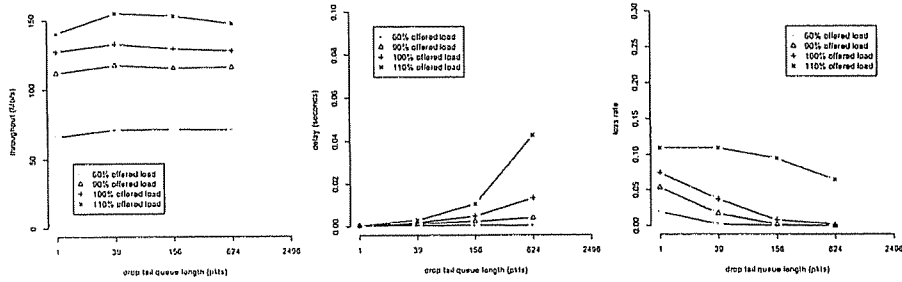


Figure 4: Aggregate profiles of mean throughput, delay, and loss for Juniper OC3 with web-like sources and drop-tail queuing discipline. As noted in Section 3.1, the Juniper M320 OC3 interface has a hard upper limit of 50 milliseconds (≈624 1500 byte packets) on buffer size.

7

As in Figure 2, Figure 5 also depicts the individual CDFs of the three performance metrics for the 5x6=30 pairs of buffer size and offered load combinations for the setup that deploys a Cisco GSR/OC3 router with drop-tail queue fed by long-lived TCP sources.

Comparing Figures 2 and 5, we note that the variability of the per flow-based performance profiles around the depicted averages in Figure 5 is much more pronounced than for the aggregate-based profiles in Figure 2. The evidence for this is provided by the CDF plots in Figure 5 (bottom row) that show in general a wide spread for the different values of a given performance metric, and typically not just for the corner cases. The practical implication is that performance profiles based on, say, 90th percentiles would deviate significantly from their average-based counterparts shown in Figure 5 (top row) and look quite different. Keeping this feature in mind when interpreting average- and per flow-based performance profiles, the main feature in Figures 5–7 is that the throughput performance profiles, when computed on a per flow basis, are no longer insensitive to changes in buffer size and/or offered load. In fact, per-flow throughput tends to decrease as offered load increases, which in turn causes an increase in per-flow loss rates. Not surprisingly, we typically also observe an increase in per-flow delay as buffer size increases. These and other observations (not shown) for certain subclasses of flows (e.g., classified by RTT or flow size) confirm the per flow-based findings reported in [13] and complement them by using our large design space to validate these main characteristics for a wider range of possible traffic, load, router, and buffer scenarios. As mentioned in [13], gaining a better understanding of the observed tradeoffs between per-flow throughput, delay, and loss is an important open issue for assessing application-layer performance, and we will revisit it in Section 5 below. The per-flow perspective also brings up issues of fairness (e.g., see [31]), but we will leave those for future work.

### 4.3 Performance Profiles: Impact of RED

In a final set of experiments that explore sensitivity aspects of the buffer sizing problem, we consider the same experimental setups as in Figure 2, but with routers that have RED queues instead of drop-tail queues, with the RED configuration settings given in Table 2. Figures 8 and 9 should be compared to Figures 2 and 5, respectively, and show the resulting RED-induced performance profile (with corresponding CDFs), computed for the aggregate (Figure 8) and on a per-flow basis (Figure 9). We observe that the resulting aggregate-based performance profiles exhibit in general only small variability around the plotted averages (as evidenced by the step-function like CDFs) and show typically a higher degree of insensitivity to differences in buffer size and/or offered traffic load for throughput and delay than their drop-tail counterparts.

In addition, RED-induced performance tends to be better than drop-tail based performance when performance is measured in terms of throughput or delay and about the same in terms of loss-based performance. With respect to the per flow-based performance profiles shown in Figure 9, RED-induced performance appears overall comparable to drop-tail based performance, with the exception of delay, with RED's delay performance profile being less sensitive and better than its drop-tail counterpart (which is to be expected, as reducing delay is one of the goals of RED). In short, without special tuning of the RED parameters and simply by relying on a set of default configuration settings, RED queues appear to result in somewhat more robust and slightly improved aggregate and per flow performance profiles when compared to their drop-tail counterparts, even in rather constrained buffer configurations. These stronger insensitivity properties of RED queues are appealing in view of SLA-related efforts to meet certain levels of performance when carrying a customer's traffic, largely irrespective of the volume and type of traffic the customer generates.

## 5 Buffer Sizing: An SLA Perspective

The empirical findings discussed in Section 4 illustrate that the "black art" of buffer sizing [13] could benefit from a new perspective, especially one that provides critical context for a range of traffic engineering issues, including configuration and provisioning decisions. Below we put forward one such perspective that focuses on Service Level Agreements. We first discuss the notion of SLAs, their purpose and engineering considerations. We then illustrate how the proposed SLA perspective can refine the router buffer sizing problem by placing it into the context of an ISP's objective to be successful in a competitive marketplace.

### 5.1 Service Level Agreements: An Overview

SLAs reflect the immense importance of IP networking to today's business enterprises and governmental institutions. A ten minute disruption of network service can cause millions of dollars of loss, or can jeopardize the functionality of essential infrastructure. SLAs spell out in a contract the technical and business relationship between network providers and customers, with positive financial consequences if the SLA is met (fees collected for services delivered) and negative ones if the SLA is not met (e.g., penalties and damage to the business relationship). SLAs for IP networks today may span the globe and multiple autonomous systems, under the control of a single network provider, a set of cooperating providers, or a network integrator. At a technical level, SLAs provide assurances on a plethora of conditions regarding connectivity, time to handle outages or close trouble tickets, and increasingly on network performance, e.g., packet delay loss
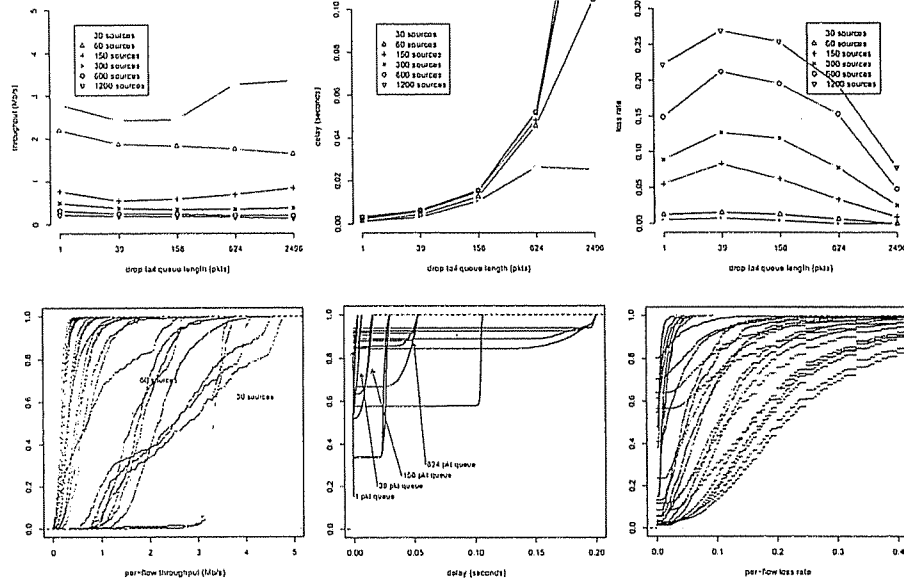
8

Figure 5: Per-flow results for Cisco OC3 with infinite sources and drop-tail queuing discipline. Profiles of mean throughput, delay, and loss shown on top row and corresponding CDFs shown along bottom row.
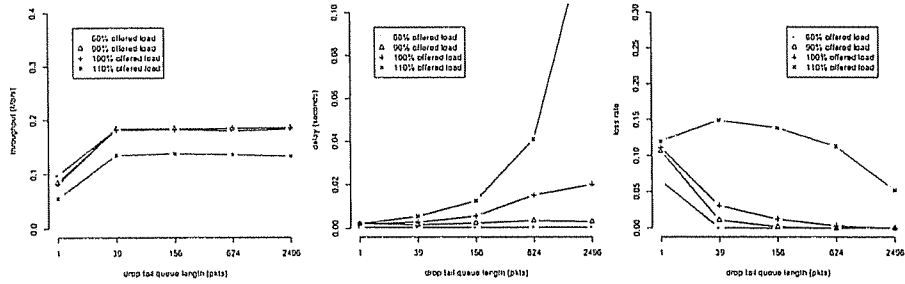


Figure 6: Per-flow profiles of mean throughput, delay, and loss for Cisco OC3 with web-like sources and drop-tail queuing discipline
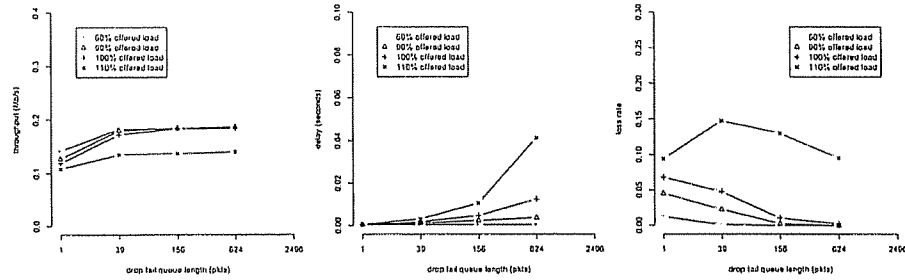


Figure 7: Per-flow profiles of mean throughput, delay, and loss for Juniper OC3 with web-like sources and drop-tail queuing discipline. As noted in Section 3.1, the Juniper M320 OC3 interface has a hard upper limit of 50 milliseconds (≈624 1500 byte packets) on buffer size.
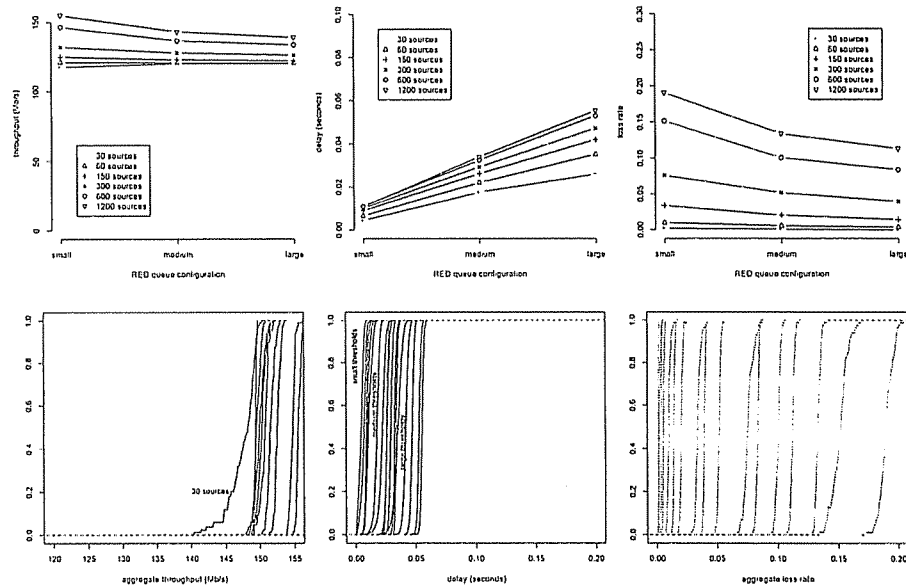
Figure 8: Aggregate results for Cisco OC3 with infinite sources and RED queuing discipline. Profiles of mean throughput, delay, and loss shown on top row and corresponding CDFs shown along bottom row.
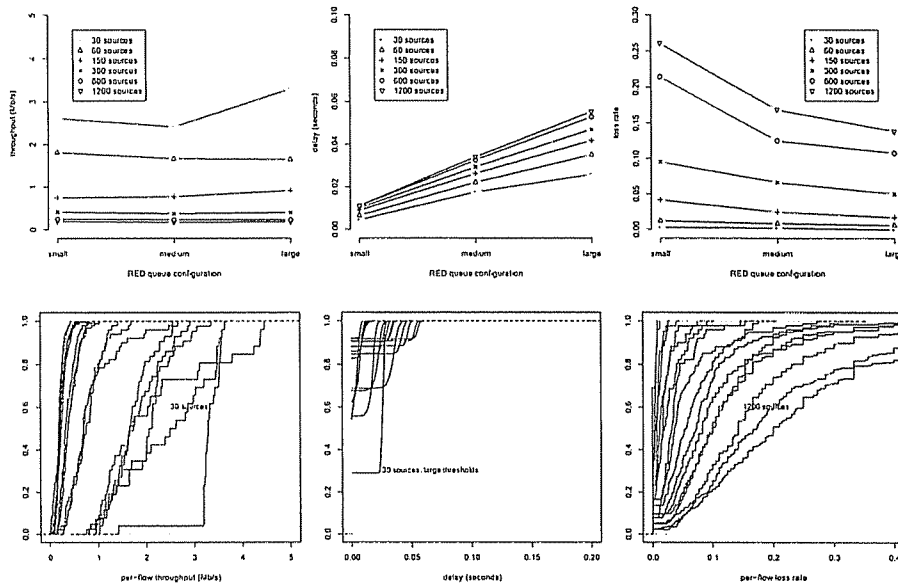


Figure 9: Per-flow results for Cisco OC3 with infinite sources and RED queuing discipline Profiles of mean throughput, delay, and loss shown on top row and corresponding CDFs shown along bottom row

10

and jitter. In this paper, we concentrate on TCP-based applications where loss and delay play a major role, but jitter is typically of lesser concern.

SLAs are of particular importance for Virtual Private Networks (VPNs). VPNs can be implemented using a wide variety of networking technologies, but all essentially provide a clear separation of the Provider Edge (PE) and Customer Edge (CE) routers or interfaces. The customer's CEs attach to the provider's PEs, and the provider's core routers provide transport between PEs. VPN services (which may be point to point or any to any among the customers CEs) thus allow customers to out-source their private network to a shared provider infrastructure. The provider manages the PE routers, core routers connecting the PEs, and (depending on the business relationship) the CE routers. The provider can support SLAs within the perimeter it controls, for example from CE to CE. Such SLAs covering network performance (*e.g.*, loss and delay) are of increasing importance to customers. Customers want assurances of little variation from agreed upon performance targets, and this is reflected in the SLAs. For example, large enterprise networks often have a hub and spoke topology (with a small number of hubs and a large number of spokes), where SLAs assure good performance from hubs to associated spokes, as well as between the hubs. Performance cannot be assured across intervals where resources are oversubscribed, and so the SLA may allow for discarding of all measurements collected when utilization exceeds a given threshold. The question is, what role does buffer sizing play in these environments?

Performance across today's large IP network cores is largely determined by two factors: (i) transmission characteristics, *i.e.*, fiber layouts and transmission rates, and (ii) PE and CE router configuration and resource management. To design and manage for SLAs, the details of core router behavior play a relatively small role, given the capacity and redundancy built into modern network cores. Transmission characteristics are readily accounted for through understanding lower layer routing and restoration capabilities. The key to engineering to meet SLA targets then quickly reduces to understanding the per hop performance characteristics of the routers on the edges, the CEs and the PEs, and this is where our experimental investigation meets SLA engineering. The problem is not easy. Network service providers have to grapple with genuine uncertainties—packet sizes, traffic mix, which even if understood in advance, may change rapidly in time

To meet customer expectations and to engineer their networks successfully, providers seek design for *robust* compliance to an SLA. Specifically, providers seek simple and *universal* rules for router hardware and software allocation, determining router buffer allocations, packet scheduling and shaping algorithms, class of service and drop priority (RED) profiles, and so forth. By robust, we mean that the configuration supports an SLA under a wide range of traffic condi-

tions. By universal, we mean that to the largest extent possible, the rules are identical for all routers in the same role (*e.g.*, CE or PE) irrespective of the details of their geographic placement in the network. Geographic details do matter in setting SLA targets of course; *e.g.*, the delay target between Shanghai and Miami. Yet, providers seek CE configuration rules that would be identical in these two cities.

## 5.2 SLAs and Buffer Sizing: Illustrative Examples

In the following, we seek to illuminate the extent to which providers can engineer networks for robust SLA compliance, and help to quantify the risk of SLA non-compliance. To this end, we consider a range of "toy" SLAs that provide assurances for network performance expressed in terms of delay and loss. While the SLAs we consider resemble real-world SLAs, they are necessarily much simpler than the latter, but we use them here mainly for illustrative purposes (for ISP-specific SLA-related information, see *e.g.*, [1, 5, 6]). In particular, our SLAs cover a range of target loss rates and target delay bounds, from reasonably tight (*i.e.*, 0.1% loss, 5 millisecond delay) to rather loose (*i.e.*, 2% loss, 50 millisecond delay), and set the link utilization threshold beyond which an SLA becomes non-binding to 70%. As a result, we eliminate some of our traffic scenarios with high offered loads, since they are intended to saturate the bottleneck link and, in turn, tend to make the SLA non-binding for most of the time.

For a given buffer size-traffic scenario pair $(X, Y)$, to measure SLA compliance, we obtain the "ground truth" by relying on the passive measurements of the traffic seen on the access link that is generated by the particular traffic scenario $Y$ and fed into router A (see Figure 1) with buffer size $X$. We imitate actual SLA reporting by slicing the passive trace data into smaller intervals (here we use 10 second segments; actual intervals are typically 5-10 minutes). For each interval, we check whether or not the utilization during that interval was below the 70% threshold. If so, we compute loss rate and (average) delay, and if not, we simply discard the interval. To obtain the final score, we compute the average of all (valid) 10 second scores across the entire trace and report SLA compliance/non-compliance depending on whether the final scores are within the target delay and loss rate thresholds. In Figure 10, each of the sixteen 2D plots corresponds to a particular SLA. For a given buffer size $X$ (x-axis) and traffic/load scenario $Y$ (y-axis), we plot a particular letter in the corresponding $(X, Y)$ coordinate if the SLA is violated in some way. A blank indicates that the SLA is met. The letter coding is explained in the caption of the figure. This coarse-grained reporting of SLA compliance is preferred by service providers because it allows for temporary SLA violations which smooth out when averaging over a large time interval (*e.g.*, week or month). Nevertheless, Figure 10 shows that from an SLA perspective, small buffer configurations are
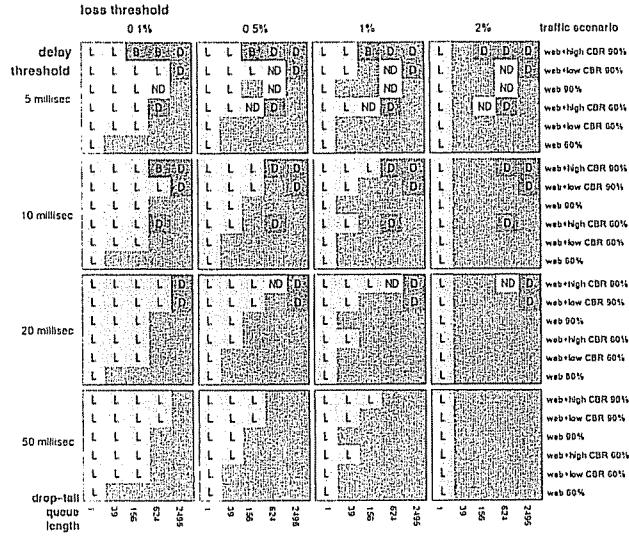
11

Figure 10: 4x4 matrix display of the scores from 16 SLAs using coarse-grained (entire trace) SLA compliance reporting (OC3, 50 millisecond round-trip time). Each SLA has a delay and loss rate threshold. Four delay thresholds (top to bottom) are used and four loss rate thresholds (left to right) are used. Each of the 16 SLAs corresponds to one of the 4x4 matrix elements, and each of them is a 6x5 block, with columns representing 5 different drop-tail queue sizes and with rows representing 6 different traffic scenarios. Scores from an individual traffic scenario/queue length combination are coded as L=loss violation, NL=near loss violation, D=delay violation, ND=near delay violation, B=both delay and loss thresholds violated, and NB=near violation of both delay and loss thresholds. "Near" means that the score is within 10% of the threshold. Blank squares indicate SLA compliance

to be avoided across the whole spectrum of traffic types and loads. Moreover, for a tight SLA (top left plot), the buffer size needs to be substantial, in which case delay is almost certain to cause problems. This unavoidable tradeoff is evident and to some degree quantified in Figure 10. Complying with a tight SLA is costly for the provider, because it can easily be violated by a number of realistic traffic scenarios, despite configuring buffers to be large. In turn, the provider will charge the customer accordingly to compensate for the loss of revenues due to providing large buffers and crediting the customer when the SLA is not met. On the other hand, since a loose SLA (bottom right plot) is relatively easy to satisfy for a wide range of possible traffic mixtures, even with moderate buffer sizes, they are less taxing for the provider and hence cheaper for the customer. SLAs with mixed constraints (top right or bottom left plot) have their own economics and their tradeoffs can be read from Figure 10.

To show the impact that the choice of time scale has on reporting SLA compliance, and by extension, on buffer sizing, Figure 11 shows the results of a fine-grained reporting of SLA compliance. Instead of averaging the scores of the (valid) 10 second slices across the entire trace, we take the individual 10 second slices and associated raw scores (i.e.,

no averaging) and plot a particular letter in the corresponding $(X, Y)$ coordinate if the SLA is violated one way or another during at least one 10 second segment. A blank indicates that the SLA is met in each (valid) 10 second interval across the whole trace. The letter coding is explained in the caption of the figure. While fine-grained SLA compliance reporting is favored by customers because it relates more directly to customer-perceived quality of service, Figures 10 and 11 depict one aspect of how this tension between what the provider prefers and what the customer desires materializes when making configuration and provisioning decisions. For one and the same SLA, the risk of SLA non-compliance is typically greater with fine-grained SLA compliance reporting than with coarse-grained and can be quantified to some degree by the increase in buffer size needed to achieve roughly the same degree of robustness (to uncertainties in traffic type and volume) of SLA compliance. In this sense, the SLA perspective captured in Figures 10 and 11 illuminates how a desire to engineer for robust SLA compliance can influence buffer sizing at the CEs and the PEs, and how buffer sizing decisions for edge routers can help to quantify the risk of SLA non-compliance, at least within the context of the large design space associated with our testbed environment. Clearly, an analytical treatment of how to engineer for robust SLA compliance and how to quantify the risk of SLA non-compliance looms as a promising open problem, but looks very daunting at this point.

## 6 Discussion and Conclusions

The landmark study by Appenzeller et al. has renewed interest in the problem of how to size and configure buffers in routers [8]. While on the surface the objective is easily stated, i.e., select the minimum size that satisfies target performance requirements, the problem has many facets which until now have typically been explored in a piecemeal manner. The first to advocate a more holistic view were Dhamdhere and Dovrolis [12], and one of the goals of our work is build on their work and broadly examine the tradeoffs between buffer configuration, traffic mix and performance. We do this by conducting a large number of experiments in a controlled laboratory setting using instances of three different commercial routers widely deployed in today's Internet. Our experiments show that all performance metrics are by and large insensitive to router architecture, and that aggregate throughput (the performance metric used in [8]) is also insensitive to buffer size and traffic mix. However, all other performance metrics show clear dependencies on buffer size, traffic mix, traffic volume and round trip time delay.

By mapping four different known buffer sizing policies into the design space established by our experiments, we are able to compare them and assess their advantages and disadvantages. In particular, this mapping highlights the perfor-
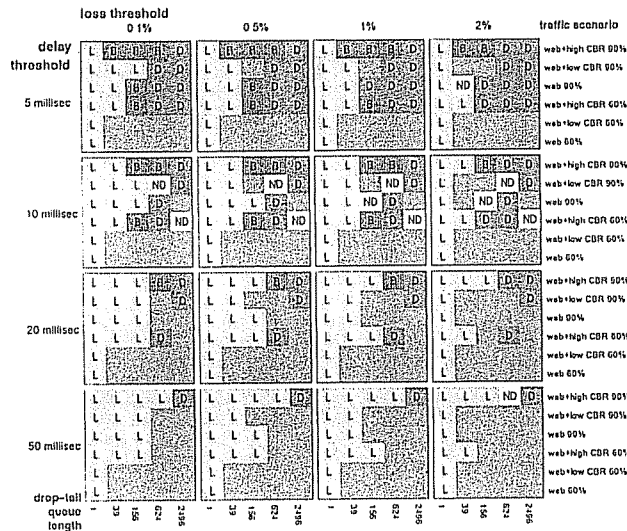
12

Figure 11: 4x4 matrix display of the scores from 16 SLAs using fine-grained (10 second time scale) SLA compliance reporting (OC3, 50 millisecond round-trip time). Each SLA has a delay and loss rate threshold. Four delay thresholds (top to bottom) are used and four loss rate thresholds (left to right) are used. Each of the 16 SLAs corresponds to one of the 4x4 matrix elements, and each of them is a 6x5 block, with columns representing 5 different drop-tail queue sizes and with rows representing 6 different traffic scenarios. Scores from an individual traffic scenario/queue length combination are coded as L=loss violation, NL=near loss violation, D=delay violation, ND=near delay violation, B=both delay and loss thresholds violated, and NB=near violation of both delay and loss thresholds. "Near" means that the score is within 10% of the threshold. Blank squares indicate SLA compliance in every (valid) 10 second interval across the entire trace.

mance risks of the buffer sizes specified by [8], and shows that the other policies either result in relatively large buffers or in buffers similar to [8], depending on traffic mix. This illustrates the need for a new perspective in support of making more informed buffer sizing decisions. Thus, a second goal of our work is to put forward a novel perspective: that of the ISP and by extension the service level agreements that drive their networks. That is, we argue that buffer sizing decisions must not only grapple with constraints of router design and network performance, but also ISP economics [11]. To this end, we consider a set of toy SLAs and use them to study the buffer sizing problem within the context of the economic incentives behind a marketplace for SLAs. In particular, we find that the type of SLA (i.e., tight, loose or in between) and how SLA compliance is monitored to be contributing factors for making informed buffer configuration decisions at the network edge. For example, tying SLA-specific performance closely to traffic mix and load opens up the possibility that careful traffic engineering may enable smaller router buffers to meet certain SLAs, and do so more effectively for coarse-grained SLA compliance monitoring. In this context, the observed benefits of AQM in small buffer configurations suggest that this often maligned technology

may play a more prominent role in networks with a thriving SLA business model.

Indeed, our results concerning performance improvements with RED warrant additional analysis. While our focus here is on qualitative analysis, statistical comparisons (e.g., tests for significance of characteristics observed in Section 4.3) also support RED as providing somewhat more robust/insensitive performance. Exploring the specific nature of these improvements, and the performance and economic tradeoffs between buffer size and management policy both empirically and analytically are subjects for future work.

One curious feature that many of the plots shown in this paper have in common is worth mentioning. It concerns the somewhat unusual property (see for example Figure 2 or to a lesser extent 3) that the aggregate loss rates in the experiments with a buffer size of 1 packet are lower than the loss rates with the buffer sizes of 39 and 156 packets. Note that in these experiments, data traffic is unidirectional and buffers in the data and ACK direction are symmetrically configured. In our initial investigation of this phenomenon, we found that in the 1 packet configuration, there was actually a fairly large amount of packet loss in the ACK direction. We conjecture that the ACK loss results in a more drastic reduction or even some kind of pacing of the data traffic resulting in lower overall loss, but plan to investigate this in more detail in future work.

While our study is designed to address a wide range of aspects of router buffer sizing, several caveats remain. First is the issue of representativeness of our experimental results. We argue that the use of commercial routers and the broad consideration of traffic and performance in our study provides an improved and broader perspective on buffer sizing. However, this does not obviate the need to refine our experimental environment or for future analytical, simulation and live deployment tests which we believe also provide useful perspectives. Furthermore, many aspects of our study could easily be adopted by router vendors and service providers who already conduct extensive tests on live systems. Finally, an appealing aspect of [8] was a closed-form expression for specifying buffer sizes. Our results demonstrate the need for expressions that include a broader range of considerations, which is the focus of our future work.

Another critical issue is that traffic characteristics are known to be quite different in the core versus the edge of the Internet. Traffic in the core tends to be relatively smooth, while traffic in the edge tends to be more bursty. It is clear that the target of the Stanford study [8] is core routers that are therefore less susceptible to traffic bursts. However, there is actually a continuum between core and edge, with the edge much larger than the core. Thus, it is important to understand the conditions under which small buffers pose a risk.

All of our tests were conducted with a two versions of TCP (from the FreeBSD 5.4 and Linux 2.6 kernels). While there are obviously a wide variety of TCP variants in the Internet

today, it is not clear that the details of differences between the dominant versions of TCP would alter our results in a meaningful way. However, we plan to examine this issue in future work. Perhaps more interesting is the question of how specific versions of TCP behave through small buffers. A first step in this direction was taken in [19] by running simulations that included both NewReno and Vegas. However, more expansive tests are desirable, including some with high-speed TCP variants.

Fairness is another consideration in assessing the impact of buffer sizing. Wischik addresses the issue of fairness in [31] and encourages consideration of other mechanisms (such as AQM) to address loss (and by extension fairness) in order to preserve the benefits of smaller buffers. While we do not address the issue of fairness directly in our study, the ability of RED to improve performance in our experiments would lend support to Wischik's argument. We plan to explore fairness in greater detail in future work.

There are also the arguments by Dukkipati and McKeown in [14] in favor of flow completion time as the "right" metric for congestion control. This raises the following important question: Should SLAs be designed to relate to FCT, and if so, how?

Last but not least, while our results serve to illuminate a wide range of known and new issues, we nevertheless believe that the problem of router buffer sizing in a competitive Internet remains largely unsolved.

# References

[1] AT&T Managed Internet Service (MIS) http://new.serviceguide.att.com/mis.htm, 2006

[2] Cisco Systems. Cisco 12000 Series Internet Router Architecture: Packet Switching http://www.cisco.com/warp/public/63/arch12000-packetsw.html, 2006

[3] Juniper Networks. Configuring the Scheduler Buffer Size http://www.juniper.net/techpubs/software/junos/junos74/swconfig74-cos/html/cos-scheduler-maps3.html, 2006

[4] Juniper Networks. Data Flow Through the M320 Router and T640 Routing Node Packet Forwarding Engine http://www.juniper.net/techpubs/software/nog/nog-hardware/html/key-components12.html, 2006

[5] NTT Communications Global IP Network Service Level Agreement (SLA) http://www.ntt.net/products/sla/sla_ts.cfm, 2006

[6] Sprint NEXTEL service level agreements http://www.sprint.com/business/support/serviceLevelAgreements.jsp, 2006

[7] S Agarwal, J Sommers, and P Barford Scalable network path emulation In *Proceedings of IEEE MASCOTS '05*, September 2005

[8] G Appenzeller, I Keslassy, and N McKeown Sizing router buffers In *Proceedings of ACM SIGCOMM '04*, Portland, OR, August 2004

[9] K Avrachenkov, U Ayesta, E Altman, P. Nain, and C Barakat The Effect of Router Buffer Size on TCP Performance In *Proceedings of the LONIIS Workshop on Telecommunication Networks and Teletraffic Theory*, St Petersburg, Russia, January 2002

[10] M Christiansen, K Jeffay, D Ott, and F Smith Tuning RED for Web Traffic In *Proceedings of ACM SIGCOMM '00*, Stockholm, Sweeden, August 2000

[11] D Clark, J Wroclawski, K Sollins, and R Braden Tussle in cyberspace: defining tomorrow's Internet *IEEE/ACM Transactions on Networking*, 13(3):462–475, 2005

[12] A. Dhamdhere, , H Jiang, and C Dovrolis Buffer Sizing for Congested Internet Links *Proceedings of IEEE INFOCOM '05*, 2005

[13] A Dhamdhere and C Dovrolis Open issues in router buffer sizing *ACM SIGCOMM Computer Communications Review*, 36(1):87 – 92, January 2006

[14] N Dukkipati and N McKeown Why flow-completion time is the right metric for congestion control. *ACM SIGCOMM Computer Communications Review*, 36(1):59 – 62, January 2006

[15] M Enachescu, A. Goel, N McKeown, and T Roughgarden Part III: Routers with Very Small Buffers *ACM SIGCOMM Computer Communications Review*, 35(3), July 2005

[16] S. Floyd. http://www.icir.org/floyd/red.html, 2006.

[17] S Floyd and V Jacobson Random Early Detection Gateways for Congestion Avoidance *IEEE/ACM Transactions on Networking*, 1(4), August 1993

[18] M. Garetto and D. Towsley Modeling, Simulation and Measurements of Queuing Delay under Long-tail Internet Traffic In *Proceedings of ACM SIGMETRICS '03*, San Diego, CA, June 2003

[19] S. Gorinsky, A Kantawala, and J Turner Link Buffer Sizing: A New Look at the Old Problem. In *Proceedings of the IEEE Symposium on Computers and Communications*, Cartagena, Spain, June 2005.

[20] Y Joo, V. Ribeiro, A Feldmann, A Gilbert, and W Willinger. TCP/IP traffic dynamics and network performance: a lesson in workload modeling, flow control, and trace-driven simulations. *ACM SIGCOMM Computer Communications Review*, 31(2), April 2001

[21] L. Le, J. Aikat, K Jeffay, and F. Smith. The Effects of Active Queue Management on Web Traffic In *In Proceedings of ACM SIGCOMM '03*, Karlsruhe, Germany, August 2003

[22] W. Leland, M Taqqu, W Willinger, and D Wilson On the self-similar nature of Ethernet traffic (extended version) *IEEE/ACM Transactions on Networking*, pages 2:1–15, 1994

[23] M. May, T Bonald, C Diot, and J. Bolot. Reasons not to Deploy RED. In *Proceedings of the Seventh IEEE International Workshop on Quality of Service*, London, England, June 1999.

[24] R Morris TCP Behavior with Many Flows. In *Proceedings of IEEE International Conference on Network Protocols*, Atlanta, GA, October 1997

[25] R Morris Scalable TCP Congestion Control In *Proceedings of IEEE INFOCOM*, April 2000.

[26] K Park, G Kim, and M Crovella On the Effect of Traffic Self-Similarity on Network Performance In *Proceedings of SPIE International Conference on Performance and Control of Network Systems*, November 1997

[27] G Raina, D Towsley, and D. Wischik Part II: Control Theory for Buffer Sizing *ACM SIGCOMM Computer Communications Review*, 35(3), July 2005

[28] A. Shaikh and A Greenberg Operations and Management of IP Networks: What Researchers Should Know Tutorial Session, ACM SIGCOMM '05 http://www.research.att.com/~albert/sigcomm05-greenberg-shaikh-tute.pdf, August, 2005

[29] J. Sommers and P. Barford Self-configuring network traffic generation. In *Proceedings of ACM SIGCOMM Internet Measurement Conference '04*, 2004.

[30] C Villamizar and C. Song High Performance TCP in ANSNET *ACM SIGCOMM Computer Communications Review*, 24(5), October 1994.

[31] D. Wischik Fairness, QoS and Buffer Sizing *ACM SIGCOMM Computer Communications Review*, 36(1), January 2006.

[32] D Wischik and N. McKeown Part I: Buffer Sizes for Core Routers *ACM SIGCOMM Computer Communications Review*, 35(3), July 2005