Buffer Sizing for Congested Internet Links

Amogh Dhamdhere, Hao Jiang, Constantinos Dovrolis
{*amogh, hjiang, dovrolis*}*@cc.gatech.edu*
Networking and Telecommunications Group
Georgia Institute of Technology

***Abstract*— Packet buffers in router/switch interfaces constitute a central element of packet networks. The appropriate sizing of these buffers is an important and open research problem. Much of the previous work on buffer sizing modeled the traffic as an exogenous process, i.e., independent of the network state, ignoring the fact that the offered load from TCP flows depends on delays and losses in the network. In TCP-aware work, the objective has often been to maximize the utilization of the link, without considering the resulting loss rate. Also, previous TCP-aware buffer sizing schemes did not distinguish between flows that are bottlenecked at the given link and flows that are bottlenecked elsewhere, or that are limited by their size or advertised window.**

In this work, we derive the minimum buffer requirement for a Drop-Tail link, given constraints on the minimum utilization, maximum loss rate, and maximum queueing delay, when it is feasible to achieve all three constraints. Our results are applicable when most of the traffic (80-90%) at the given link is generated by large TCP flows that are bottlenecked at that link. For heterogeneous flows, we show that the buffer requirement depends on the harmonic mean of their round-trip times, and on the degree of loss synchronization. To limit the maximum loss rate, the buffer should be proportional to the number of flows that are bottlenecked at that link, when that number exceeds a certain threshold. The maximum queueing delay constraint, on the other hand, provides a simple upper bound on the buffer requirement. We also describe how to estimate the parameters of our buffer sizing formula from packet and loss traces, evaluate the proposed model with simulations, and compare it with two other buffer provisioning schemes.

Keywords: TCP, congestion control, queue and buffer management, routers and packet switches.

I. Introduction

The packet buffers of router or switch interfaces ("links") are an essential component of packet networks. They absorb the rate variations of incoming traffic, delaying packets when there is contention for the same output link. In general, increasing the buffer space at a router interface tends to increase the link utilization and decrease the loss rate, but it also tends to increase the maximum queueing delay. Consequently, a simple but fundamental question is: *what is the minimum buffer requirement of a network link given certain constraints on the minimum utilization, maximum loss rate, and maximum queueing delay?*

Perhaps surprisingly, the answer to this question is still not well understood, and several different answers are often quoted. Typically, a router or switch interface will be configured at the time of its purchase with a certain amount of buffering. A major vendor recommends that a router interface should have 500ms worth of buffering, implying that the buffer requirement is proportional to the capacity of the corresponding link. But why is this the right answer, and if so, why is 500ms the right delay? A common rule of thumb is that the buffer requirement is equal to the bandwidth-delay product, in which the bandwidth term is the capacity of the link, and the delay term corresponds to the Round-Trip Time (RTT) of a TCP connection that may saturate that link. However, what if the link is loaded with several TCP flows? What if their RTTs are different? And how do different types of flows (e.g., large versus small) affect the required buffer size?

The buffer sizing problem would be tractable if we had an accurate model to characterize Internet traffic. In that case, it might be possible to calculate the buffer requirement, given certain performance objectives, based on an analytical or numerical evaluation of the corresponding queueing system. Significant advances have been made in that direction, especially in the context of QoS provisioning; we refer the reader to [1], [2] and to the references therein. Such approaches, however, face two major problems. First, an accurate and parsimonious traffic model that would be valid for any Internet link is still to be found, while the accurate parameterization of such models is often an even harder process. Second, more importantly, the previous approach is intrinsically "open-loop", meaning that the traffic that is applied to a queue is an *exogenous process* that does not depend on the state of the network. Clearly that is not the case with Internet traffic, given that TCP flows, which account for almost 90% of Internet traffic, react to packet losses and RTT variations. A closed-loop approach, on the other hand, would couple the traffic sources with the amount of buffering and the resulting delays and losses in the network, adjusting the throughput of the former based on how TCP behaves.

A well cited paper in the area of buffer sizing for TCP traffic is the work by Villamizar and Song [3]. That is an early experimental study, performed at a WAN testbed, that measured the link utilization and the throughput of end-to-end bulk TCP transfers with different buffer configurations. A recommendation given by [3] is that the buffer space should be at least equal to the bandwidth-delay product of the link, where the "delay" term refers to the RTT of a single TCP flow that attempts to saturate that link. No recommendations

This work was supported by the NSF CAREER award ANIR-0347374 and by an equipment donation from Intel Corporation.

are given, however, for the more realistic case of multiple TCP flows with different RTTs.

Quite recently, Appenzeller et al. [4] concluded that the buffer requirement at a link decreases with the square root of the number N of TCP flows that are active at that link. According to their analysis, the buffer requirement to achieve almost 100% utilization is

$$B = \frac{CT_{avg}}{\sqrt{N}} \tag{1}$$

referred to as the "Stanford scheme" henceforth. Note that for N=1, this is the rule of thumb in [3]. The key insight behind the Stanford scheme is that, when the number of competing flows is sufficiently large, they can be considered independent and non-synchronized, and so the standard deviation of the aggregate offered load (and of the queue occupancy) decreases with the square root of N. An important point about the Stanford model is that it aims to keep the utilization close to 100%, without considering the resulting loss rate. We believe that the loss rate is a crucial performance metric in TCP/IP networks, and that buffer sizing should aim to keep the loss rate bounded to a small value. Furthermore, the Stanford scheme is applicable when N is so large that the effects of (even partial) loss synchronization on buffer sizing can be ignored.

The first work to consider the effect of the number of competing TCP flows on the buffer sizing problem was by Morris in [5], [6]. He recognized that the loss rate increases dramatically with the number of active TCP flows, and that buffering based on the bandwidth-delay product can be grossly insufficient in practice, causing frequent TCP timeouts and unacceptable variations in the throughput and transfer latency of competing TCP transfers [5]. He also proposed the Flow-Proportional Queueing (FPQ) mechanism, as a variation of RED, which adjusts the amount of buffering based on the number of active TCP flows. FPQ is a cornerstone of our work, and we explain it in detail in §IV. However, neither [6] and [4] distinguish between flows that are bottlenecked at the given link and flows that are bottlenecked elsewhere, or between flows that are limited by their size or advertised window. We show that the buffer requirement of a link depends *not* on the number of active flows, but *on the number of flows that are throughput-limited due to congestion at that link*.

A related area of work is that of Active Queue Management (AQM) [7], [8], [9], [10]. Instead of using simple Drop-Tail queues, AQM uses early drops, before buffer overflows occur, aiming to control the average queue size independent of the physical buffer size, stabilize the queue size, and avoid bursty losses and global loss synchronization. Note that AQM schemes cannot control the maximum loss rate. Also, the effects of heterogeneous RTTs, non-persistent connections, and of the number of competing flows on AQM parameters are not well understood. Furthermore, AQM schemes are not deployed in the Internet, at least so far. Even though our buffer sizing method is not directly applicable to AQM schemes, it is important to note that AQM schemes would not solve the main problem that we consider in this paper (namely, to maintain full utilization with maximum loss rate and queueing delay constraints).

In this work, we focus on the buffer requirement of a Drop-Tail queue given constraints on the minimum utilization, maximum loss-rate, and, when feasible, on the maximum queueing delay. Specifically, we derive the *minimum* buffer size that is required so that the link can be fully utilized by heterogeneous TCP flows, while keeping the loss rate and the queueing delay bounded by given thresholds. The minimum amount of buffering to satisfy these constraints is preferred, because larger buffers cost more and they lead to increased queueing delays and jitter. We show that the buffer requirement given N heterogeneous TCP flows depends on the *harmonic mean* of their round-trip times[1]. We also show that the degree of loss synchronization can significantly affect the buffer requirement, especially for links that carry less than a few tens of persistent TCP flows at a time (for instance, access or edge links). To limit the loss rate at the link, we show that the minimum buffer requirement is proportional to the number of flows N_b that are "bottlenecked" at that link, i.e., throughput-limited only by local congestion, when N_b exceeds a threshold. Depending on the value of N_b and the given delay bound, it may not be feasible to satisfy both the loss rate and delay constraints. In that case, the operator can perform buffer sizing based on the constraint that he/she considers as most important.

Our model is applicable and accurate when most of the traffic (80-90%) at the given link belongs to the N_b locally bottlenecked flows. The rest of the traffic can be UDP flows, short TCP flows, window-limited TCP flows, and persistent TCP flows that are bottlenecked elsewhere. The main contribution of the paper is a buffer sizing formula that we refer to as *Buffer Sizing for Congested Links (BSCL)*. We also describe how to estimate the parameters that BSCL depends on, namely flow RTTs, number of locally bottlenecked flows, and a loss synchronization factor, from packet and loss traces. Finally, we use simulations to validate BSCL and to compare it with the bandwidth-delay product buffer sizing formula, and with the Stanford scheme.

Paper structure: Section II describes our link and traffic model and states the buffer sizing problem. Section III derives the buffer requirement focusing only on the utilization constraint. Section IV extends the previous result considering the maximum loss rate constraint. Section V introduces the maximum delay constraint and gives the condition under which the buffer sizing problem has a solution. Section VI describes how to estimate the parameters of BSCL. Section VII validates BSCL and compares it with two related schemes. We conclude in Section VIII.

[1] The harmonic mean of N numbers $T_1, \ldots T_N$ is given by $\frac{N}{\sum \frac{1}{T_i}}$.

II. Traffic model and objectives

Our model of a router/switch interface is a single queue with constant capacity C bytes/sec and buffer space B bytes. The queue follows the Drop-Tail policy, meaning that it drops an arriving packet if there is not enough space in the buffer. Of course a modern router interface is much more complex than this simple queue. Our model is applicable, however, under the following assumptions. First, we consider output-queued multiplexers, in which packets are switched to the appropriate output interface before any significant queueing at the input ports. In router architectures that use virtual-output-queueing or shared memory among all interfaces, the model should be adjusted to consider a variable capacity C or a variable buffer space B, respectively. Second, the assumption of a single queue per interface is justified by the fact that, even though many routers provide several queues for different classes of service, typically only one of them is used. Third, we assume that the buffer is structured in terms of bytes, rather than packets or cells of a certain size. Modifying the results for buffers with a more coarse addressing granularity should be straightforward.

The buffer sizing objectives that we consider are related to three major performance metrics for a network link: utilization, loss rate, and queueing delay. Specifically, we want to calculate the amount of buffering B that satisfies the following constraints, when it is feasible to do so:

1) **Full utilization**: The average utilization ρ of the link should be at least $\hat{\rho} \approx 100\%$ when the offered load is sufficiently high. In the rest of the paper, we set $\hat{\rho}$=98%.
2) **Maximum loss rate** $\hat{p}$: The loss rate p should not exceed $\hat{p}$, typically 1-2% for a saturated link. A limited loss rate allows persistent TCP connections to achieve a significant throughput, and it reduces the throughput variability among different flows. A low loss rate is also important for short TCP flows, that often cannot recover from losses using Fast-Retransmit, and for interactive or real-time applications that cannot afford retransmission delays. Morris showed simulation results for the transfer latency of short TCP flows as the loss rate increases [6]. His results show that the loss rate increases almost with the square of the number of competing flows. The loss rate increase causes, not only a reduction in the average flow throughput, but also a significant increase in the variation of the transfer latency across different flows. This is because some short flows are "lucky" and they do not see packet losses, while other flows experience several losses and retransmission timeouts. In other words, a large loss rate causes significant unfairness in the bandwidth distribution among flows, and also large transfer latencies due to frequent TCP retransmission timeouts.
3) **Maximum queueing delay** $\hat{d}$: The queueing delay d should not exceed a bound $\hat{d}$. Even though a queueing delay requirement can be stated in terms of short-term averages, or in terms of the delay tail distribution, a bound on the maximum queueing delay is simpler to verify and it leads to deterministic, rather than statistical, guarantees. The SLAs provided by major ISPs today are often expressed in terms of maximum delays. The maximum delay requirement is important for real-time applications, and it is also related to the transfer latency of short TCP flows. In general, increasing B tends to increase ρ, decrease p, but at the same time increase d. This implies that *the given constraints* ($\hat{\rho}, \hat{p}, \hat{d}$) *may not be feasible*. Specifically, for the maximum delay requirement to be met, we must have that $B < C\hat{d}$. It is possible however that unless if B is larger than $C\hat{d}$, it is not feasible to meet utilization and/or loss rate constraints. In that case, the operator would have to choose whether the maximum delay constraint is more important than the utilization and loss rate constraints, and perform buffer sizing accordingly. We return to this issue in §V.

In the following, we refer to the link that we focus on as the "target link". The answer to the buffer sizing problem is intimately related to the traffic at the target link. We next describe various types of traffic.

a) **Locally Bottlenecked Persistent (LBP) TCP flows***:* These are large TCP flows which are only limited, in terms of throughput, by congestive losses at the target link. The throughput R of LBP flows can be approximated by the following simple formula, derived in [11],

$$R = \frac{0.87M}{T\sqrt{p}} \tag{2}$$

where M is the flow's Maximum Segment Size, T is the flow's average RTT, and p is the loss rate that the flow experiences. The previous model is fairly accurate when p is less than 2-5% and most losses are recovered with Fast-Retransmit. More accurate models, taking into account retransmission timeouts, a maximum advertised window, or different variations of TCP (such as SACK) exist in the literature (see [12], [13] and references therein). We prefer to use (2), however, mostly due to its simplicity. It is important to note that, for LBP flows, the loss rate p should be equal to the loss rate at the target link, i.e., the flow should not encounter losses elsewhere in its path. Also note that the average window of an LBP flow, $W = RT$, is independent of the flow's RTT.

b) **Remotely Bottlenecked Persistent (RBP) TCP flows***:* These are also large TCP flows that are only limited by congestion, and their throughput can be approximated by (2). The difference with LBP flows is that RBP flows also experience losses in links other than the target link, and that their throughput bottleneck is not the target link. Hence the value of p for these flows in (2) would be *larger* than the loss rate at the target link.

c) **Window-limited Persistent TCP flows***:* These are also large TCP flows, but their throughput is limited by the receiver advertised window. Note that window-limited flows may also experience packet losses at the target link. Their difference with LBP flows is that the latter keep increasing their window until a loss occurs.

d) **Short TCP flows and non-TCP traffic***:* Finally, the traffic at the target link may include TCP mice and non-TCP flows. The former spend most of their lifetime in slow start, and they typically account for a small share of the aggregate traffic in Internet links [14], [15]. The latter can be viewed as exogenous traffic, and they are also typically a small share.

The key assumption in the rest of this paper is that most of the traffic at the target link is generated from LBP flows. The reason behind this assumption is that we heavily rely on the "sawtooth" behavior of TCP flows, as well as on (2). Note that non-LBP traffic could also contribute to the buffer requirement. Our conjecture is, however, that if the LBP flows account for almost the entire aggregate traffic, then any additional buffer requirement due to non-LBP flows can be ignored. The simulations of § VII show that the proposed BSCL formula is valid as long as LBP flows account for 80-90% of the aggregate traffic in the target link.

The major implication of the previous assumption is that BSCL will be mostly applicable in edge and access networks, where certain links can become congested with large TCP flows that are locally bottlenecked. Core network links, or links that rarely become the bottleneck for the majority of their traffic, should not be provisioned based on BSCL. For such links, the Stanford scheme of [4] may be more appropriate. On the other hand, the Stanford scheme can lead to a high loss rate in links with a large fraction of LBP traffic, as shown in § VII. We note that even though most of the Internet links today are probably well-provisioned and not congested, there are certainly bottlenecks at "last-mile" links, access networks, as well as in the developing world.

III. Utilization constraint

Consider, initially, the utilization constraint $\hat{\rho} \approx$ 100%, and let us assume that the traffic consists only of N_b LBP flows[2]. The objective of buffer provisioning in this case is to "hide" the TCP window reductions due to congestive losses from the link's output rate. This can be achieved by buffering enough traffic before the losses, so that even after any congestive losses and window reductions the link remains saturated.

A. Single flow

In the case of a single TCP flow with RTT T seconds, the previous insight leads to the well-known *bandwidth-delay product* buffer sizing formula $B \geq CT$. In detail, suppose that the flow has reached a window size W^{max} when it causes a buffer overflow. At that time the window is $W^{max} \geq CT + B$, and so one or more packets must be dropped. In congestion avoidance, in particular, only one packet will be dropped because $W^{max} = CT + B + 1$. If all dropped packets are detected with Fast-Retransmit as a single congestion event, which is more likely to happen if the connection uses the SACK option, the window will be reduced by a factor of two and the window will become $W^{min} = W^{max}/2$. So, the minimum window size after a congestion event can be as low as $W^{min} = (CT + B)/2$. The link will remain saturated as long as $W^{min} \geq CT$, and so the minimum required buffer space is

$$B = CT \tag{3}$$

The previous buffer requirement has the form of a "bandwidth-delay product", with "bandwidth" referring to the capacity of the link and "delay" referring to the RTT of the TCP connection that saturates the link.

Two remarks about this formula: First, an important assumption is that the window is reduced by only a factor of two. However, in slow-start, and especially if SACK is not used, several packets can be lost at the same time causing multiple window reductions, or even timeouts. In that case, (3) may not be sufficient to avoid a utilization drop. Second, the RTT term of (3) does not include any queueing delays in the target link; however, queueing delays in other links should be included in T.

B. Heterogeneous flows - global synchronization

Consider now the general case of N_b heterogeneous LBP flows with RTT T_i, i=$1, \dots N_b$. We first derive the minimum buffer requirement for full utilization considering the worst-case scenario in which all flows experience a loss at the same congestion event. Such *global loss synchronization* events are common in Drop-Tail buffers carrying just a few flows.

Suppose that during a particular congestion event around time t_c each of the N_b flows reduces its window from a maximum value W_i^{max} to a minimum value W_i^{min}. As in the case of a single flow, we assume that after the congestion event the window of each flow is reduced by a factor of two, i.e., $W_i^{min} = W_i^{max}/2$. Before the congestion event, the backlog of flow i at the buffer is $Q_i^{max} = W_i^{max} - W_i^{flt}$, where W_i^{flt} is the amount of bytes from flow i "in-flight" in the path, but not backlogged at the queue of the target link, just prior to the congestion event. As shown in [12], the expected value of the TCP congestion window at any point in time does *not* depend on the flow's RTT. So, as long as all N_b LBP flows experience the same loss rate at the target link, we have that W_i^{flt}=W^{flt} for all i.

The aggregate backlog before the congestion event is

$$Q^{max} = \sum_{i=1}^{N_b} (W_i^{max} - W^{flt}) \tag{4}$$

Since all the windows are at their maximum values, this is the maximum possible backlog at the bottleneck link. To ensure that the link remains saturated after t_c, when the windows are at their minimum values, we must have that

$$Q^{min} = \sum_{i=1}^{N_b} (W_i^{min} - W^{flt}) \geq 0 \tag{5}$$

[2] We use N_b for the number of LBP flows to distinguish from the total number of flows N.

where Q^{min} is the backlog at the bottleneck after t_c. To derive the *minimum* buffer requirement, we consider the extreme case in which W_i^{min}=W^{flt} for all i, and so Q^{min}=0. So, since $2W_i^{min} = W_i^{max}$, the maximum possible backlog is given by

$$Q^{max} = N_b W^{flt} \tag{6}$$

In order for the N_b flows to saturate the link even when the aggregate backlog is zero, we must have that

$$\sum_{i=1}^{N_b} R_i^{min} = C \tag{7}$$

where R_i^{min} is the throughput of flow i after the congestion event,

$$R_i^{min} = \frac{W_i^{min}}{T_i} = \frac{W^{flt}}{T_i} \tag{8}$$

From (7) and (8), we get that

$$R_i^{min} = \frac{C}{T_i \sum_{i=1}^{N_b} 1/T_i} \tag{9}$$

and so the maximum possible backlog follows from (6)

$$Q^{max} = N_b W^{flt} = \sum_{i=1}^{N_b} R_i^{min} T_i = \sum_{i=1}^{N_b} \frac{C}{\sum_{i=1}^{N_b} 1/T_i} \tag{10}$$

We set the minimum buffer requirement to be the maximum possible backlog i.e. $B = Q^{max}$. Hence, the minimum buffer requirement for saturating the link can be written again as a bandwidth-delay product

$$B = CT_e \tag{11}$$

where T_e is referred to as the *effective RTT* of the N_b flows, and it is given by the *harmonic mean* of their RTTs,

$$T_e = \sum_{i=1}^{N_b} \frac{1}{\sum_{i=1}^{N_b} 1/T_i} = \frac{N_b}{\sum_{i=1}^{N_b} 1/T_i} \tag{12}$$

In the case of N_b homogeneous flows with RTT T_i=T, the effective RTT is equal to T, and the minimum buffer requirement becomes as in the case of a single flow.

It is interesting that the effective RTT is given by the harmonic mean, as opposed to the arithmetic mean of the flow RTTs. The harmonic mean of N_b uniformly distributed positive values is lower than their arithmetic mean. For instance, suppose that we have N_b RTTs uniformly distributed from $\frac{T}{N_b}$ to T i.e. $T_i = \frac{Ti}{N_b}$ with $i = 1, \ldots N_b$. The effective RTT is given by $T_e = \frac{T}{\sum_{i=1}^{N_b} 1/i}$, which can be approximated by $T_e \approx T/(lnN_b + 1)$. For N_b=1000, we get $T_e \approx T/8$, while the RTT arithmetic mean is approximately four times larger.

The reason that the buffer requirement depends more heavily on small RTTs is that the corresponding flows have a larger share of their window backlogged in the buffer of the target link than elsewhere in the path, compared to flows with larger RTTs (remember that they all have the same average window). A practical implication of this result is that the buffer requirement can remain relatively small, even when a few connections have atypically large RTTs, as long as most LBP flows have small RTTs. This would not be the case if the effective RTT was determined by the arithmetic mean.

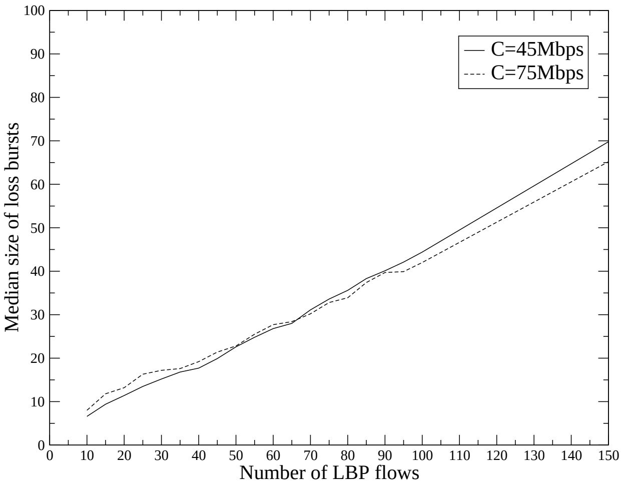

Fig. 1. Loss-burst length as a function of the number of flows.

C. Heterogeneous flows - partial synchronization

The assumption of global loss synchronization gave us a simple result for the minimum buffer requirement, but it is fairly restrictive. We now derive the buffer requirement in the case of *partial loss synchronization* with a simple probabilistic model.

The starting point for this model is the relation between the number of LBP flows N_b and the *loss-burst length* $L(N_b)$ i.e., the number of dropped packets during a congestion event. A congestion event can be loosely defined as a time period in which one or more packets from different flows are dropped in a short duration, relative to the duration of loss-free time periods.

To empirically measure $L(N_b)$ with simulations, we examine the time series of the packet drops at the target link and count the number of successive drops that are spaced by less than the effective RTT of the corresponding flows. The reason for choosing the RTT as the appropriate time scale is because TCP flows detect and react to packet losses in (roughly) a single RTT. We observed, through simulations, that $L(N_b)$ tends to increase with the number of LBP flows N_b. To illustrate this, Figure 1 shows simulation results for the median loss-burst length as a function of N_b[3]. Notice that the loss-burst length increases as the number of LBP flows increases. This is because, as N_b increases, more flows experience packet drops every time the buffers of that link overflow, even if the flows are not completely synchronized. Also note that $L(N_b)$ increases almost linearly with N_b with a slope that is less than

[3] More details about the simulation topology and parameters are given in the Appendix. For the results of Figure 1, we have C=45Mbps and 75Mbps, with B=CT_{avg} packets. These simulations were run ten times with different random seeds, and Figure 1 shows the median loss-burst length across the ten runs.

one, at least for the range of N_b in Figure 1; we return to this point later in the section.

If we could estimate the average size of a loss-burst $\bar{L}_{N_b}$ for N_b LBP flows, then we can derive the minimum buffer requirement for full utilization. Consider a congestion event with the average loss-burst length $\bar{L}_{N_b}$. The probability that none of the $\bar{L}_{N_b}$ dropped packets belongs to a particular flow i is $(1-1/N_b)^{\bar{L}_{N_b}}$. So, the fraction of flows that are expected to see at least one loss is

$$q(N_b) = 1 - (1 - \frac{1}{N_b})^{\bar{L}_{N_b}} \qquad (13)$$

Note that, the fraction $q(N_b)$ decreases as N_b increases, meaning that it becomes less likely for a given flow to see a loss during a congestion event.

Suppose that W is the average window size (in bytes) of a flow *before* a loss-burst of length $\bar{L}_{N_b}$. After the loss-burst, we expect that a fraction $q(N_b)$ of flows will see losses and they will reduce their window by a factor of two, while the rest of the flows will increase their window by one packet. Thus, the average window size W' after the congestion event will be

$$W' = q(N_b)\frac{W}{2} + [1 - q(N_b)](W + M) \qquad (14)$$

where M is the flow's segment size. The link was saturated before the congestion event, when the buffer was full, and so $N_b W \geq CT + B$. We require that the link stays saturated after the congestion event. Since we are interested in the minimum buffer requirement, we can consider the case that the buffer becomes empty after the congestion event, and so $N_b W' = CT$. This is equivalent to the following expression

$$q(N_b)\frac{CT+B}{2N_b} + [1 - q(N_b)](\frac{CT+B}{N_b} + M) = \frac{CT}{N_b} \qquad (15)$$

Solving for B, the minimum buffer requirement is

$$B = \frac{q(N_b)CT - 2MN_b[1 - q(N_b)]}{2 - q(N_b)} \qquad (16)$$

where $q(N_b)$ can be calculated from (13) if we have an estimate of the average size of a loss-burst for the given number of flows N_b.

Some remarks on (16) follow. First, in the case of global loss synchronization $q(N_b)$=1 and so the resulting buffer requirement becomes $B = CT$, as in (11) for the case of homogeneous flows. Second, the effect of partial loss synchronization is to reduce the buffer requirement, since B decreases as N_b increases. Intuitively, this is because when flows are partially synchronized, they do not reduce their windows at exactly the same time, and so the amount of backlogged traffic that is needed to keep the link saturated is reduced. Third, the above equation is derived considering N_b *homogeneous* LBP flows. To take into account heterogeneous connections, we would replace T in the above equation with T_e, where T_e is the effective RTT defined earlier.

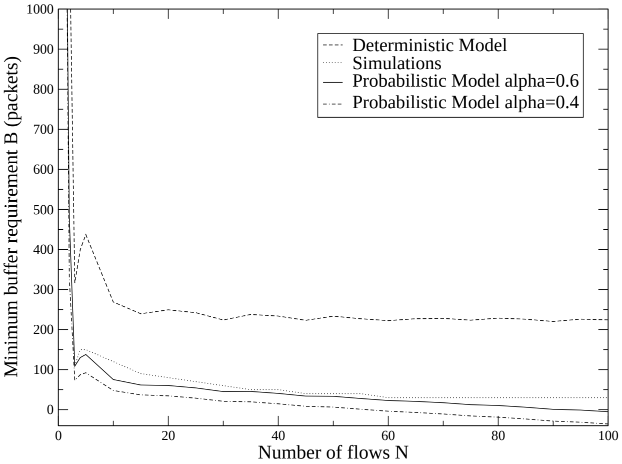

Fig. 2. The deterministic model (11), the probabilistic model (16), and simulation results for a utilization constraint $\hat{\rho}$=98%. Note that the TCP flows are heterogeneous, and so their effective RTT changes as we increase N. This explains the spike of the curves when $N \approx 5$.

D. Model validation

Figure 2 compares the deterministic model (11), the probabilistic model (16), and simulation results for the minimum required buffer size that keeps the utilization above $\hat{\rho}$=98%. Note that the flows have different RTTs (see Appendix), and so the effective RTT T_e varies as we increase N_b.

To simplify the estimation of $\bar{L}_{N_b}$, let us go back to Figure 1. Observe that the relation between loss-burst length and N_b is almost linear, and so we can approximate $\bar{L}_{N_b}$ as

$$\bar{L}_{N_b} \approx \alpha N_b \qquad (17)$$

where α is the *loss synchronization factor*. Our simulations have shown that α <1, typically in the range 0.4-0.7 for the range of N_b shown in Figure 2. We observed that the linear approximation (17) remains valid as long as N_b is more than 3-5 flows, and less than a couple of hundreds of flows. As will be shown in the next section, the buffer requirement depends on the loss synchronization factor only when N_b is less than a few tens of flows. For larger values of N_b, the buffer requirement is determined by the maximum loss rate constraint and it does not depend on $\bar{L}_{N_b}$. So, the linear approximation of (17) is sufficient for our purposes.

Figure 2 shows that the probabilistic model (16) follows closely the simulation results for a loss synchronization factor α=0.6. With α=0.4, which was observed in the simulations of Figure 1, (16) underestimates the buffer requirement. The error is not large, however, implying that the buffer requirement is robust to errors in the estimate of α. Notice that the deterministic model $B = CT_e$, on the other hand, overestimates the required buffer size by a factor of almost ten for large values of N_b. This overestimation can lead to large queues, and increased transfer latencies and jitter. Thus, we prefer to use the probabilistic model, even though it requires a rough estimate for the loss synchronization factor α (or for $\bar{L}_{N_b}$).

Note that Figure 2 shows the buffer requirement in terms of packets, while (16) is in terms of bytes. To convert from bytes to packets, the buffer size given by (16) is divided by the segment size M=1500B.

The probabilistic model remains accurate as long as the number of flows is less than about 80 in the simulations of Figure 2. For a larger number of flows, the loss rate becomes higher than 4-5%, flows often experience timeouts and multiple window reductions during the same congestion event, and our probabilistic model becomes inaccurate. In that operating region, however, the buffer requirement does not depend on the utilization constraint, but on the maximum loss rate constraint that we examine in the next section. So, the probabilistic model (16) is accurate for the operating region in which it is meant to be used.

IV. Loss rate constraint

The previous section derived the minimum buffer requirement for saturating a link that is loaded with LBP flows. Even though utilization is a major concern for a network operator, the service that end-users perceive can be quite poor when the loss rate is more than about 5%. Losses are particularly detrimental for relatively short and interactive flows, such as HTTP flows, which often have to recover dropped packets with retransmission timeouts [16]. Even for bulk transfers, a high loss rate can cause timeouts to those "unlucky" flows that experience multiple nearby losses, affecting them more severely than other flows [5]. Consequently, we are also interested in an upper bound $\hat{p}$ on the loss rate of a congested link. In the rest of the paper, we assume that $\hat{p}$ is 1%.

There is an intimate connection between the loss rate p at a link and the number N_b of LBP flows at the target link. From (2), we see that if N_b homogeneous flows with RTT T saturate a link with capacity C, the loss rate p will be proportional to the square of N_b,

$$p = {N_b}^2 (\frac{0.87}{CT})^2 \tag{18}$$

The previous expression is accurate only when the link is adequately buffered so that the N_b flows can saturate it. Also, (18) is valid as long as the model of (2) is applicable; if p is larger than about 3-5%, timeouts become common and p increases almost linearly with N_b [5].

One way to interpret (18) is that, given an upper bound $\hat{p}$ on the loss rate, the number of persistent TCP flows should be less than $\sqrt{\hat{p}}CT/0.87$. That would require, however, an admission control scheme, limiting the number of flows that can access the network. Such schemes have been previously proposed [17], but not deployed.

A. Flow Proportional Queueing

Another approach, referred to as *Flow Proportional Queueing (FPQ)* [6], is to meet the loss rate constraint $\hat{p}$, not by limiting N_b, but by increasing the RTT of the flows as N_b increases. Since the RTT includes the queueing delay at the target link, increasing the buffer space of that link would increase the RTTs of the carried TCP flows. This would tend to keep the loss rate constant in spite of an increase in the number of TCP flows.

Specifically, suppose that the N_b flows are homogeneous, and that their RTT is

$$T = T_p + T_q \tag{19}$$

where T_q is the queueing delay at the target link and T_p accounts for all other delays along the forward or reverse paths of the flows. From (18) and (19), we see that the queueing delay $\hat{T}_q$ that is required to meet the loss rate constraint $\hat{p}$ is

$$\hat{T}_q = \frac{0.87}{C\sqrt{\hat{p}}} N_b - T_p \tag{20}$$

The key observation is that *to keep the loss rate constant, the queueing delay should increase proportionally to N_b.*

The average queueing delay T_q in a congested link can be assumed to be, as a first-order approximation, equal to the maximum queueing delay B/C. Then, the buffer requirement to keep the loss rate below $\hat{p}$ is

$$B \approx C\hat{T}_q = K_p N_b - CT_p \tag{21}$$

where

$$K_p = \frac{0.87}{\sqrt{\hat{p}}} \tag{22}$$

So, B has to be sufficiently large so that each flow can have a window of K_p packets, either stored in the buffer (B term) or elsewhere in the path (CT_p term). Note that K_p≈9 packets for $\hat{p}$=1%, and K_p≈6 packets for $\hat{p}$=2%.

The buffer sizing formula (21) is basically the same as the FPQ scheme of [6]. Comparing (21) with FPQ (see Figure 5 of that reference), we see that FPQ sets K_p to 6 packets, and it does not take into account the term CT_p. Another difference is that [6] proposes to have K_p packets per *active* TCP flow at the target link. However, in our model, we only take into account flows that are *bottlenecked* at the target link (LBP flows).

B. Integrated model for $\hat{\rho}$ and $\hat{p}$

(16) was derived considering only the utilization constraint $\hat{\rho}$, while (21) only considered the loss rate constraint $\hat{p}$. To meet both constraints, B has to be sufficiently large to satisfy the most stringent of the two requirements for any value of N_b. Since (21) increases proportionally with N_b, we expect that B will be determined by the utilization constraint if N_b is less than a certain number of flows $\tilde{N}_b$ (if the effective RTT remains constant). If N_b>$\tilde{N}_b$, B is determined by the loss rate constraint instead.

In the case of heterogeneous flows with different RTTs, the term T_p in (21) should be replaced with the effective RTT T_e of (12). The reason is that the number of bytes that are "in-flight" in the path, but not stored in the buffer, is

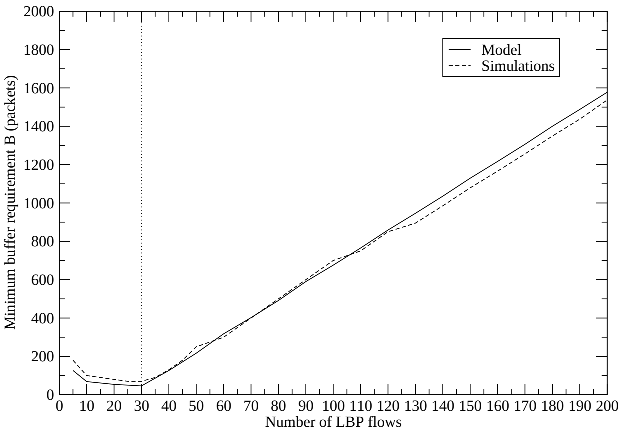

Fig. 3. Analytical and simulation results for $\hat{\rho}$=98% and $\hat{p}$=1%.

CT_e. Thus, the minimum buffer requirement for the given constraints $\hat{\rho}{\approx}100\%$ and $\hat{p}$ is

$$\hat{B} = \begin{cases} B_\rho = \frac{q(N_b)CT_e - 2MN_b[1-q(N_b)]}{2-q(N_b)} & \text{if } N_b < \tilde{N}_b \\ B_p = K_pN_b - CT_e & \text{if } N_b > \tilde{N}_b \end{cases} \tag{23}$$

where $\tilde{N}_b$ is the value of N_b at which B_ρ=B_p, $q(N_b)$ is given by (13), the effective RTT T_e is given by (12), and K_p is given by (22).

Figure 3 compares the buffer sizing formula (23) with simulation results for the minimum buffer requirement when $\hat{\rho}$=98% and $\hat{p}$=1%. We assume that the loss synchronization factor is α=0.55. Note that the flows have different RTTs (heterogeneous case), and so the effective RTT T_e changes as we vary N_b. $\tilde{N}_b$ is about 30 flows, which is shown by the dotted line in Figure 3. We see that the model slightly underestimates the buffer requirement, due to the probabilistic nature of loss synchronization or due to an error in the chosen value of α. The error is, however, quite small. For a larger number of flows, the minimum buffer requirement given by the simulations increases almost linearly with N_b, and (23) is quite accurate in matching both the slope and magnitude of the required buffer. Further validation results, considering also some limited amount of traffic from non-LBP flows, are given in § VII.

V. Delay constraint

The previous two sections considered the utilization and loss rate constraints, which are both important for throughput-intensive data transfer applications. Many applications, however, also have end-to-end delay constraints. For instance, real-time conferencing, IP telephony, and telnet-like applications can tolerate a maximum delay before their utility drops to zero. As discussed in §II, even though the delay constraint can be stated in a probabilistic manner based on the delay tail distribution, a maximum delay constraint $\hat{d}$ is easier to work with and it does not depend on the statistical characteristics of the traffic.

Web browsing flows, which are TCP-based, can also be viewed as interactive [18]. Since they typically consist of only a few data segments, their transfer latency is primarily dependent on their RTTs and loss rate, and not on the capacity of the path. [16] showed how to predict the transfer latency of a short TCP flow taking into account the connection establishment phase, slow-start, and potential losses during those phases, given the RTT and loss rate. So, limiting the queueing delay at the links a Web flow goes through also limits the flow's RTT, providing a lower transfer latency.

In this section, we extend the buffer provisioning model of §IV with a maximum queueing delay constraint $\hat{d}$. Since the maximum queueing delay at a link of capacity C and buffer space B is given by B/C, the queueing delay constraint requires that B is limited by

$$B \leq C\hat{d} \tag{24}$$

We now face a *feasibility problem*: the constraints for $\hat{\rho}$, $\hat{p}$, and $\hat{d}$ will not be satisfiable if the minimum buffer requirement of (23) is larger than $C\hat{d}$. So, given C, T_e, and N, the maximum delay constraint that can be met is

$$\hat{d} > \max\{\frac{B_\rho}{C}, \frac{B_p}{C}\} \tag{25}$$

where B_ρ and B_p are as given by (23). If a lower delay bound is required, the network operator would need to sacrifice the utilization and/or loss rate objectives, limit the maximum number of persistent flows through an admission control unit, or increase the capacity C of the link.

In the rest of this paper, we assume that the utilization and loss rate requirements are more important than the maximum delay requirement. So, whenever $\hat{d}$ is not feasible, we are interested in the *minimum* buffer requirement to meet the $\hat{\rho}$ and $\hat{p}$ constraints, so that at least we minimize the maximum queueing delay. In the evaluation section, we assume that the delay constraint $\hat{d}$ is large enough so that (25) is true.

VI. Parameter estimation

In the last three sections, we derived an approximation for the minimum buffer size B that is required to saturate a link of capacity C and to meet a maximum loss rate constraint and a maximum queueing delay constraint. Figure 4 summarizes that result, which we refer to as *Buffer Sizing for Congested Links (BSCL)*. Notice that the BSCL formula requires the following traffic parameters: an estimate of the number of LBP flows (N_b), their aggregate capacity share (C_e), RTTs (T_i), and a typical Maximum Segment Size (M). We also need an estimate of the loss synchronization factor α, or an estimate of the average loss-burst length $\bar{L}_{N_b}$ for the given number of LBP flows. In this section, we describe how to estimate these parameters from passively collected traces of packet departures and loss events. Even when the previous parameters cannot be estimated accurately in practice, it is still important to know the factors that the buffer requirement depends on, and their relative impact on B. In § VII, we present some simulation

$$B = \max\{B_\rho, B_p\} \tag{26}$$

where:

$B_\rho = \frac{q(N_b)C_eT_e - 2MN_b[1-q(N_b)]}{2-q(N_b)}$

$B_p = K_pN_b - C_eT_e$

N_b: number of LBP flows at target link

C_e: effective capacity for LBP flows
(e.g., C_e=0.9C for 10% non-LBP traffic)

$T_e = \frac{N_b}{\sum_{i=1}^{N_b} 1/T_i}$: effective RTT of LBP flows

M: Maximum Segment Size for LBP flows

$K_p = 0.87/\sqrt{\hat{p}}$ (9 packets for $\hat{p}$=1%)

$q(N_b) = 1 - (1 - 1/N_b)^{\bar{L}_{N_b}}$

$\bar{L}_{N_b} \approx \alpha N_b$: Average size of a loss burst
where α is the loss synchronization factor

Fig. 4. BSCL formula: minimum buffer requirement to saturate a link of capacity C and limit the loss rate to less than $\hat{p}$.

results for the robustness of BSCL to estimation errors in N_b, α, and T_e.

Estimation of N_b and C_e: As previously mentioned, N_b is the number of large TCP flows that are limited, in terms of throughput, only due to congestive losses at the target link. In particular, we need to distinguish LBP flows from TCP flows that are bottlenecked in other links, or that are limited by their size or advertised window. Actually, the problem of classifying TCP flows based on their primary rate-limiting factor was studied in depth in [19]. That work developed techniques that determine if a TCP flow is limited by congestion, sender/receiver maximum window, "opportunity" (i.e., size), etc, based on a passively collected trace of the flow's packets. We use the techniques of [19] to detect size-limited and window-limited TCP flows[4].

To distinguish between flows that are limited by congestion at the target link (LBP) from flows that experience congestion in other links, we developed a new heuristic. The basic idea relies on the temporal correlation between the packet losses and rate reductions of a TCP flow, as observed at the target link. If a rate reduction of flow X is not preceded in the recent past by a packet loss by that flow at the target link, we infer that the window reduction must have been a consequence of a packet loss in some other link. If that happens several times, the flow is bottlenecked elsewhere (RBP). Otherwise, if most of the significant rate reductions of X follow packet losses at the target link, we count the flow as LBP. C_e is estimated as the aggregate rate of LBP flows.

Notice that window-limited TCP flows would also show strong temporal correlations between loss events and rate reductions; the key difference with LBP flows is that the latter keep increasing their window until a loss occurs. The simulations of § VII show that BSCL is robust to a limited overestimation in the number of LBP flows. So, the possible mis-classification of window-limited flows as LBPs has no major impact.

Estimation of T_e: To calculate the effective RTT of LBP flows, we need an estimate of their RTTs T_i. There are several passive measurement algorithms that provide reasonably accurate RTT estimates for TCP connections [20], [21], [22]. In this paper, we adopted the algorithm presented in [21], mostly because it does not require both traffic directions and is quite simple. The estimation technique of [21] is based on the time spacing of the 3-way-handshake TCP messages, and/or on the time spacing between the packets of the first two round-trips of a flow during slow-start. In either case, a single measurement is produced, reflecting the RTT at the start of the flow, before the latter builds up any queueing at the target link. According to [21], their estimation technique provides an RTT for 55-85% of the TCP workload, while about 90% of the measurements are accurate within 10% or 5ms, whichever is larger. The technique of [21] has a higher estimation coverage and accuracy for large TCP flows, which makes it quite appropriate for LBP flows.

Estimation of $\bar{L}_{N_b}$ or α: The B_ρ term depends on the degree of loss synchronization, which in turn depends on the average length of a loss burst $\bar{L}_{N_b}$ for that value of N_b, or, if we assume the linearity of (17), on the loss synchronization factor α. In section § III, we described how to measure the average loss burst $\bar{L}_{N_b}$ from a loss-event trace, and how to estimate α from (17). If a loss-event trace is available, then the average or median size of a loss burst $\bar{L}_{N_b}$ should be used directly, as it is more accurate especially when N_b <10 or so. However, if a loss trace is not available, then a typical value of α can be used instead, such as α=0.55.

VII. Evaluation

We evaluated the BSCL scheme using NS-2 simulations. The simulation topology is shown in the Appendix, with the capacity of the target link set to 50Mbps. We load the target link with 4 types of traffic: persistent TCP flows that are bottlenecked at the target link (LBP flows), persistent TCP flows that are bottlenecked at the access link prior to the target link (RBP flows), window-limited TCP flows, and short TCP flows. The number of RBP flows and window-limited flows is 20 and 10, respectively, while the number of LBP flows varies from 2 to 400. The short flows have an average size of 14 packets, and their interarrivals are exponentially distributed.

[4] We found, however, that the techniques described in [19] are not too accurate to identify window-limited fbws when queueing delays are comparable to propagation delays.

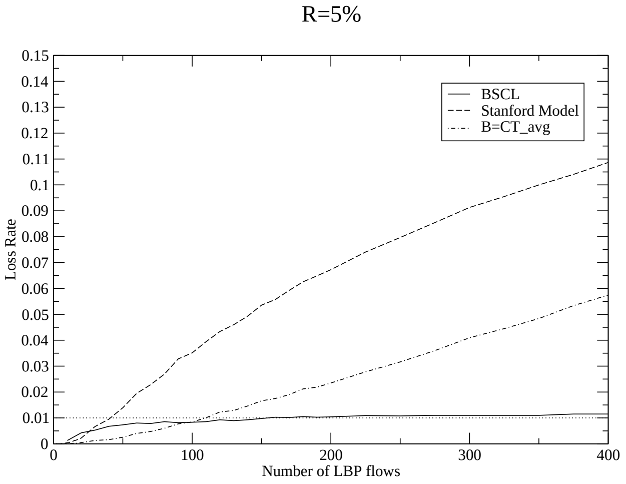

Fig. 5. Loss rate for C_e=95% of C.

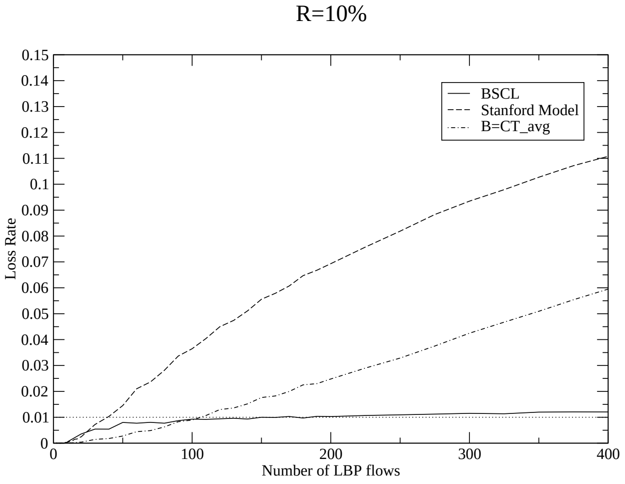

Fig. 6. Loss rate for C_e=90% of C.

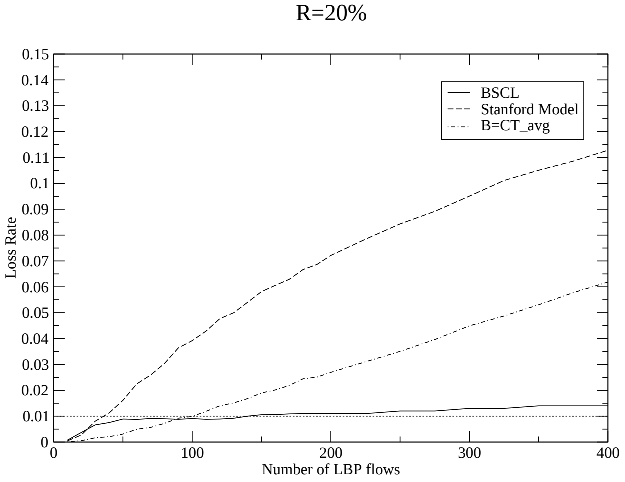

Fig. 7. Loss rate for C_e=80% of C.

The total amount of non-LBP traffic varies between 5% to 20% (i.e., C_e varies from 95% to 80% of C). 40% of the non-LBP traffic comes from RBP flows, 40% from window-limited flows, and 20% from short flows. The offered load from short flows is controlled by their interarrival time, while the offered load from RBP flows and window-limited flows is controlled by the access link capacity and receiver window, respectively. All flows start at random times in the first 10% of the simulation duration, which is then ignored from the analysis of the simulation results.

We run each simulation once with the target link buffer set to $B = CT_{avg}$, which is the rule of thumb described in § III. We then apply the estimation techniques described in § VI to get estimates for N_b, T_e and $\bar{L}_{N_b}$. From the formula of Figure 4, we then determine the BSCL requirement for a maximum loss rate $\hat{p}$=1%, and run the simulation again with that value of B. We also simulate with the buffer size of the Stanford scheme [4], which is $B = \frac{CT_{avg}}{\sqrt{N}}$. Note that [4] does not differentiate between LBP flows and other flows; for comparison purposes, we set N to N_b in their formula.

Figures 5-7 show the loss rate that is measured at the target link with each buffering scheme, when the LBP capacity share C_e decreases from 95% to 80%. We see that when C_e is 90% of C, BSCL manages to keep the loss rate close to the $\hat{p}$=1% threshold. As the amount of non-LBP traffic is increased, however, BSCL starts violating that objective. This is because we have assumed that the buffer requirement of non-LBP traffic can be ignored, which of course is not true when that share of the workload is significant compared to LBP traffic. The CT rule of thumb and the Stanford scheme, on the other hand, produce a loss rate that increases significantly with the number of LBP flows. For 300 LBP flows, the loss rate is about 4% and 10% with the CT formula and with the Stanford scheme, respectively.

We also investigated the target link utilization that results from the previous three buffering schemes (CT_{avg}, Stanford, BSCL). Simulation results (not included here due to space constraints) show that the utilization remains at 100% with all three schemes, as long as the number of LBP flows is more than 10-20. With lower values of N_b, as Figure 8 shows, BSCL can slightly underestimate the buffer requirement if $\bar{L}_{N_b}$ is estimated as the median, rather than the average, loss-burst length. This is because, at low values of N_b, we occasionally see a few very long loss-bursts that make the average loss-burst significantly longer than the median. Figure 8 also shows that BSCL causes a more significant underutilization (80%) when N_b is just two flows. This is because our probabilistic model for partial loss synchronization cannot capture the almost deterministic synchronization patterns that emerge in that case. A simple correction would be to modify BSCL so that $B = CT_e$ if N_b is less than 5 flows.

Table I shows the buffer requirement (in terms of 1500-B packets) predicted from each of the previous three buffer sizing schemes for increasing values of N_b. These results refer

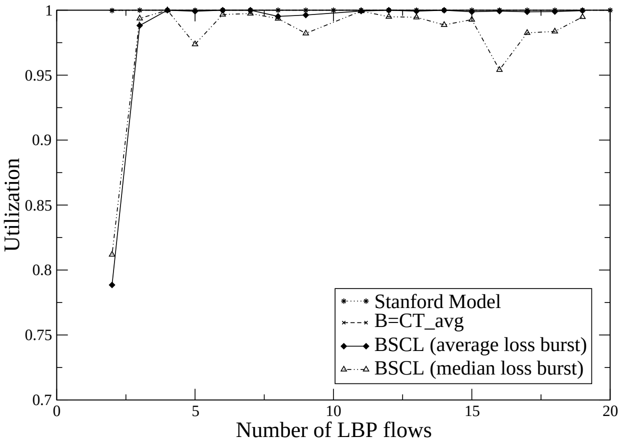

Fig. 8. Utilization for small values of N_b (C_e=90% of C). Note that the waviness in the BSCL curve is due to variations in the effective RTT of the simulated flows as we increase N_b.

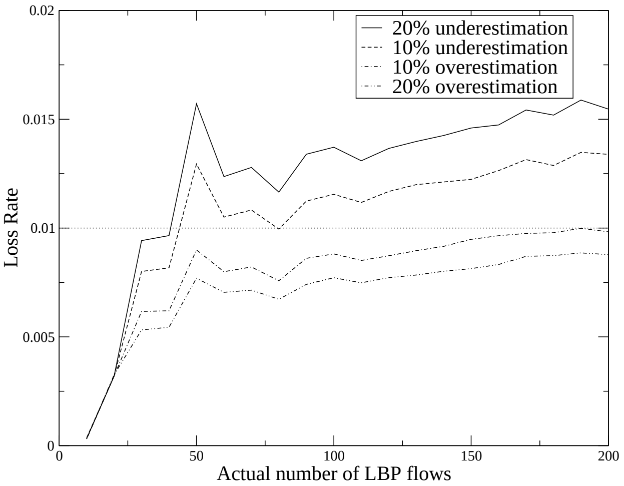

Fig. 9. Robustness to N_b estimation errors. Note that the waviness of the curves is due to variations in the effective RTT of the simulated flows as we increase N_b.

N_b	CT_{avg}	Stanford model	BSCL
2	1600	1131	336
7	1159	438	233
13	953	264	196
15	850	219	100
17	881	213	93
20	838	187	115
40	767	121	282
60	673	86	396
80	714	79	583
100	710	71	688
120	672	61	898
140	696	58	1092
160	696	55	1281
180	672	50	1508
200	689	48	1659
250	687	43	2078
300	672	38	2459
350	682	36	2882
400	682	34	3200

TABLE I

BUFFER REQUIREMENT OF THREE SIZING SCHEMES FOR INCREASING NUMBER OF LBP FLOWS.

to the case C_e=0.9×C. Note that T_e varies as we increase N_b. When N_b becomes larger than $\tilde{N_b}$=17 flows, the BSCL buffer requirement is determined by the maximum loss rate rather than the utilization constraint. The rule of thumb $B = CT$ predicts a buffer size that is much larger than that of BSCL when N_b is less than about 20-40 flows. For more flows, that rule requires less buffering than BSCL, but as previously shown, it also leads to a significant loss rate. The Stanford scheme requires more buffering than BSCL when the number of LBP flows is less than about 20, mostly because it does not consider the effect of partial loss synchronization. For more flows, the Stanford buffer requirement drops rapidly, but it also causes a significant loss rate.

It is also important to examine the robustness of BSCL to estimation errors in the parameters N_b, $\bar{L}_{N_b}$, and T_e, especially when real-time measurement and adjustment of these parameters is not feasible in practice. In our robustness study, we introduced controlled errors in N_b, $\bar{L}_{N_b}$ and T_e, after the initial estimation of the latter as described in § VI. Then, we repeated the simulations several times with the erroneous estimates to observe the impact of those errors in the loss rate and utilization at the target link. We found that overestimation or underestimation errors up to 20% in $\bar{L}_{N_b}$ or in T_e do not cause violations in the utilization or loss rate constraints. However, as Figure 9 shows, the *underestimation* of N_b by 20% does violate the loss rate constraint by up to 0.5% for N_b less than 200 flows. Overestimation of N_b by a certain factor, on the other hand, causes overestimation of the buffer requirement by the same factor, when the buffer requirement is determined by the loss rate constraint.

VIII. CONCLUSIONS

The buffer sizing problem for routers and switches has been considered "black art" for some time. Buffer provisioning based on open-loop traffic models ignores the reactive nature of TCP traffic, which accounts for at least 90% of the Internet traffic today. On the other hand, provisioning based on a typical bandwidth-delay product can lead to either poor utilization and high loss rate, or significant overestimation of the buffer requirement. The recently proposed "Stanford" scheme [4] has been a significant step forward, showing that the buffer requirement can be much lower than the bandwidth-delay product. This is a major relief for hardware designers of backbone routers, given that an OC-792 interface would otherwise require almost a Gigabyte of SRAM buffer space. It is important to understand, however, that the Stanford scheme focuses only on utilization, ignoring the resulting loss rate. With a large number of LBP flows, that buffer sizing approach can lead to a high loss rate and poor performance for many

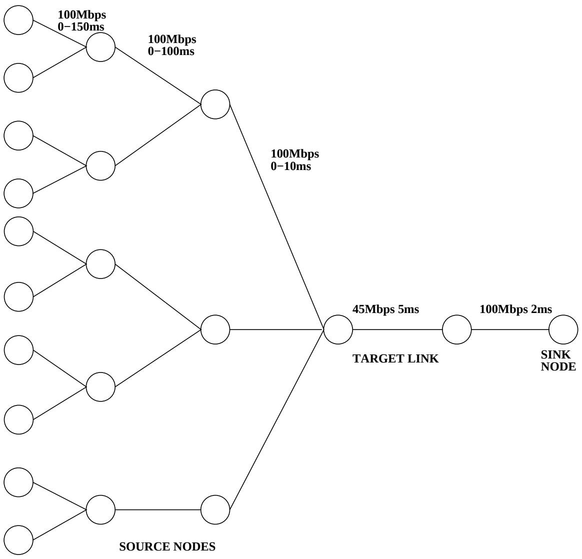

Fig. 10. Simulation topology.

applications.

The main contribution of our paper was to derive a buffer sizing formula (BSCL) for congested links. BSCL is applicable for links in which 80-90% of their traffic comes from large TCP flows that are locally bottlenecked. In that case, BSCL manages to keep the target link saturated without letting the loss rate exceed a given bound (typically 1%). BSCL considers heterogeneous RTTs and partial loss synchronization, two effects that have been largely ignored in previous work. Also, BSCL distinguishes between the total number of active TCP flows and the number of LBP flows. The former can be orders of magnitude higher than the latter in practice, because of the large number of short TCP flows. A limitation of our work is that the BSCL validation is strictly based on simulations. An interesting task for future work will be to apply the BSCL model in a congested router interface, carrying real Internet traffic, observe the resulting utilization and loss rate, and compare with other proposed buffer provisioning schemes.

Appendix: Simulation details

We have experimented with several simulation topologies. The results reported in this paper are based on the simulation topology of Figure 10. A set of 18 source nodes, located at the nodes of a tree, generate TCP flows destined to a single sink node. The internal links have a diverse set of propagation delays, causing an RTT distribution between 20ms and 534ms with an average of 217ms. The TCP flows use NewReno, Delayed ACKs, the SACK option, 1500-byte data packets, and 10,000-packet advertised windows (unless if noted otherwise). The non-bottlenecked links are provisioned in terms of both capacity and buffer size so that they do not cause losses or significant queueing delays. All simulations run for 200 seconds. Longer simulations do not produce significantly different results. The reported results ignore the first 20 seconds of each simulation run.

References

[1] J. Cao and K. Ramanan, "A Poisson Limit for Buffer Overflow Probabilities," in *Proceedings of IEEE INFOCOM*, June 2002.

[2] I. C. Paschalidis and S. Vassilaras, "Model-Based Estimation of Buffer Overflow Probabilities from Measurements," in *Proceedings of ACM SIGMETRICS*, June 2001.

[3] C. Villamizar and C.Song, "High Performance TCP in ANSNET," *ACM Computer Communication Review*, Oct. 1994.

[4] G. Appenzeller, I. Keslassy, and N. McKeown, "Sizing Router Buffers," in *Proceedings of ACM SIGCOMM*, Oct. 2004.

[5] R. Morris, "TCP behavior with many flows," in *Proceedings IEEE International Conference on Network Protocols*, Oct. 1997, pp. 205–211.

[6] ——, "Scalable TCP Congestion Control," in *Proceedings of IEEE INFOCOM*, Apr. 2000.

[7] S. Floyd and V. Jacobson, "Random Early Detection Gateways for Congestion Avoidance," *IEEE/ACM Transactions on Networking*, vol. 1, no. 4, pp. 397–413, Aug. 1993.

[8] L. Le, J. Aikat, K. Jeffay, and F. D. Smith, "The Effects of Active Queue Management on Web Performance," in *Proceedings of ACM SIGCOMM*, Aug. 2003.

[9] V. Misra, W. B. Gong, and D. Towsley, "Fluid-based Analysis of a Network of AQM Routers Supporting TCP Flows with an Application to RED," in *Proceedings of ACM SIGCOMM*, Sept. 2000.

[10] T. J. Ott, T. V. Lakshman, and L. H. Wong, "SRED: Stabilized RED," in *Proceedings of IEEE INFOCOM*, Apr. 1999.

[11] M. Mathis, J. Semke, and J. Madhavi, "The Macroscopic Behavior of the TCP Congestion Avoidance Algorithm," *ACM Computer Communications Review*, vol. 27, no. 3, pp. 67–82, July 1997.

[12] J. Padhye, V.Firoiu, D.Towsley, and J. Kurose, "Modeling TCP Throughput: A Simple Model and its Empirical Validation," in *Proceedings of ACM SIGCOMM*, 1998.

[13] B. Sikdar, S. Kalyanaraman, and K. S. Vastola, "Analytic Models for the Latency and Steady-State Throughput of TCP Tahoe, Reno and SACK," *IEEE/ACM Transactions on Networking*, vol. 11, no. 6, pp. 959–971, Dec. 2003.

[14] S. Bhattacharyya, C. Diot, J. Jetcheva, and N. Taft, "POP-Level and Access-Link-Level Traffic Dynamics in a Tier-1 POP," in *Proceedings of the ACM SIGCOMM Internet Measurement Workshop (IMW)*, Nov. 2001.

[15] N. Brownlee and K. Claffy, "Internet Stream Size Distributions," in *SIGMETRICS '02: Proceedings of the 2002 ACM SIGMETRICS international conference on Measurement and modeling of computer systems*. ACM Press, 2002, pp. 282–283.

[16] N. Cardwell, S.Savage, and T.Anderson, "Modeling TCP Latency," in *Proceedings of IEEE INFOCOM*, Mar. 2000.

[17] R. Mortier, I. Pratt, C. Clark, and S. Crosby, "Implicit Admission Control," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 12, pp. 2629–2639, Dec. 2000.

[18] P. R. Selvidge, B. Chaparro, and G. T. Bender, "The World Wide Wait: Effects of Delays on User Performance," in *Proceedings of the IEA 2000/HFES 2000 Congress*, 2000.

[19] Y. Zhang, L. Breslau, V. Paxson, and S. Shenker, "On the characteristics and origins of internet flow rates," in *Proceedings of ACM SIGCOMM 2002*. ACM Press, 2002, pp. 309–322.

[20] M. Allman, "A Web Server's View of the Transport Layer," *Computer Communication Review*, vol. 30, no. 5, Oct. 2000.

[21] H. Jiang and C. Dovrolis, "Passive Estimation of TCP Round-Trip Times," *ACM Computer Communication Review*, Aug. 2002.

[22] S. Jaiswal, G. Iannaccone, C. Diot, J. Kurose, and D. Towsley, "Inferring TCP Connection Characteristics Through Passive Measurements," in *Proceedings of IEEE INFOCOM*, Mar. 2004.