Routers With Very Small Buffers: Anomalous Loss Performance for Mixed Real-Time and TCP Traffic

Arun Vishwanath and Vijay Sivaraman
School of Electrical Engineering and Telecommunications
University of New South Wales
Sydney, NSW 2052, Australia
Emails: {arunv@ee.unsw.edu.au, vijay@unsw.edu.au}

***Abstract*— The past few years have seen researchers debate the size of buffers required at core Internet routers. Much of this debate has focused on TCP throughput, and recent arguments supported by theory and experimentation suggest that few tens of packets of buffering suffice at bottleneck routers for TCP traffic to realise acceptable link utilisation. This paper introduces a small fraction of real-time (i.e. open-loop) traffic into the mix, and discovers an anomalous behaviour: In this specific regime of very small buffers, losses for real-time traffic do not fall monotonically with buffer size, but instead exhibit a region where larger buffers cause higher losses. Our contributions pertaining to this phenomenon are threefold: First, we demonstrate this anomalous loss performance for real-time traffic via extensive simulations including real video traces. Second, we provide qualitative explanations for the anomaly and develop a simple analytical model that reveals the dynamics of buffer sharing between TCP and real-time traffic leading to this behaviour. Third, we show how various factors such as traffic characteristics and link rates impact the severity of this anomaly. Our study particularly informs all-optical packet router designs (envisaged to have buffer sizes in the few tens of packets) and network service providers who operate their buffer sizes in this regime, of the negative impact investment in larger buffers can have on the quality of service performance.**

I. Introduction

The topic of correctly sizing buffers at core Internet routers has generated much debate in the past few years. The rule-of-thumb commonly used by router manufacturers today is attributed to [1], and requires a router to have sufficient buffers to prevent underflows that result in idling and wastage of link bandwidth. Specifically, a router should have enough buffers such that when the buffer overflows, causing TCP to react by reducing its transmission rate, there are enough packets stored to keep the output link busy, thereby ensuring that the link capacity is not wasted when TCP is increasing its transmission rate. Mathematically, this translates to a buffer size of $B = T \times C$, where T denotes the average round-trip time (RTT) of a TCP flow through the router, and C the capacity of the bottleneck link. For typical $T = 250$ ms, a router with a C = 40 Gbps link would require 10 Gigabits of buffering, which poses a considerable challenge to router design.

Theoretical analysis and practical work have recently questioned the use of the rule-of-thumb. Researchers from Stanford showed in 2004 that when a large number N of long-lived TCP flows share a bottleneck link in the core of the Internet, the absence of synchrony among the flows permits a central limit approximation of the buffer occupancy, and $B = T \times C/\sqrt{N}$ packets of buffering suffice to achieve near-100% link utilisation [2]. A core router carrying $10,000$ TCP flows needs only $10,000$ packet buffers instead of one million as governed by the earlier rule-of-thumb.

In 2005, the Stanford researchers presented theoretical and empirical evidence for further reduction in router buffers, claiming that under certain assumptions, as few as 20-50 packet buffers suffice to provide acceptable link utilisation for TCP traffic [3], [4], [5]. The claim was supported by their experiments in Sprint ATL, and also by other groups at Verizon Communications and Lucent Technologies [6], while a measurement study on a Sprint backbone router also found the queue size to seldom exceed 10 packets [7]. These initial results show the feasibility of building routers with very few packet buffers if the network can be operated at 80-90% utilisation. Clearly, the aforementioned results have significant implications from an all-optical router design point of view, where buffering presents a very important but difficult operation, since data is to be retained in the optical domain.

Very recently, in late 2007, the work in [8] revisited the ongoing buffer sizing debate from a completely different perspective. Rather than focusing purely on link utilisation, it focuses on the average per-flow TCP throughput. The authors present evidence to suggest that the output/input capacity ratio at a router's interface largely governs the amount of buffering needed at that interface. If this ratio is greater than one, then the loss rate falls exponentially, and only a very small amount of buffering is needed, which corroborates with the results reported in[4]. However, the concern is that, if the output/input capacity ratio is lesser than one, then the loss rate follows a power-law reduction and significant buffering is needed. Other concerns regarding the implications of the above buffer sizing recommendation have also been reported. The work in [9] shows that such a reduction in buffer size can lead to network instability, where instability is referred to as periodic variations in the aggregate congestion windows of all TCP flows. [10] and [11] argue that very small buffers can cause significant losses and performance degradation at the application layer. The latter presented experimental results to validate their claim. These concerns have since been partially addressed in [6] and [12], while our own prior work [13] has considered the impact of small buffers on the performance of real-time traffic.

A. Motivation

From the observation of traffic in the Internet core, it is widely accepted that nearly 90-95% of it is TCP traffic, while UDP accounts for about 5-10%. To the best of our knowledge, this has led all previous work to largely ignore the impact of very small buffers on UDP's performance. In this paper, we focus our attention on buffer sizing when both TCP and UDP (open-loop) traffic coexist in the network and

show why it is important to address the joint performance. We use the term real-time, UDP, and open-loop traffic interchangeably.

To understand the dynamics of buffer occupancy at a bottleneck link router, we mixed a small fraction of UDP traffic along with TCP and measured the UDP packet loss and end-to-end TCP throughput. Before starting our simulations, our intuition was that in the regime of very small buffers (up to 50 packets):

1) UDP packet loss would fall monotonically with buffer size, and
2) End-to-end TCP throughput would increase with buffer size to saturation as well.

Surprisingly, our observation was contrary to our intuition. We found that there exists a certain continuous region of buffer sizes (typically in the range of about 8-25 packets) wherein the performance of real-time traffic degrades with increasing buffer size. In other words, packet loss for real-time traffic increases as the buffer size increases within this region. We call this region of buffer size an "anomalous region" with respect to real-time traffic. More surprisingly, we found that when there are a sufficiently large number of TCP flows, this performance impact on UDP traffic is not at the expense of a significant improvement in end-to-end TCP throughput. On the contrary, the anomalous loss results in only a marginal increase in end-to-end TCP throughput. The inflection point occurs around the buffer size region corresponding to when TCP has nearly attained its saturation throughput.

This phenomenon is interesting for a number of reasons and forms the motivation for the research in this paper. Firstly, as real-time multimedia applications such as online gaming and interactive audio-video services continue to become more prevalent in the Internet, which is expected to increase the fraction of Internet traffic that is UDP, the anomaly suggests that the study of router buffer sizing should consider the presence of real-time traffic, and not ignore it completely.

Secondly, in this regime of very small (all-optical) buffers, it is prudent to size router buffers at a value that balances the performance of both TCP and UDP traffic appropriately. Operating the router buffers at a very small value can adversely impact the performance of both TCP and UDP traffic. Furthermore, operating it in the "anomalous region" can result in increased UDP packet loss, with only a marginal improvement in end-to-end TCP throughput.

Finally, it is known that all-optical routers can potentially offer several advantages; among them, high capacity and low power consumption [5]. However, in order to build all-optical routers, buffering of packets needs to be accomplished in the optical domain. This remains a complex and expensive process. It has been shown in [14] that emerging integrated photonic circuits can at best buffer a few dozen packets. Spools of fibre can implement fibre delay lines (FDLs) that provide optical buffering capability [15]. Unfortunately, the high speed of light implies that even minimal buffering requires large fibre spools (1 km of fibre buffers light for only 5μsec). In addition, incorporating FDLs into a typical optical switch design (such as the shared memory architecture [16]) requires larger optical crossbars, which can add significantly to the cost as the FDL buffers increase. It is therefore expected that all-optical routers will have buffering of only a few dozen packets, and the anomaly revealed by our study shows that the investment made in deploying larger buffers within this regime can negatively impact quality of service and lead to worse performance.

The rest of this paper is organised as follows. In Section II, we introduce the anomalous loss behaviour using real video traffic traces. In Section III, we provide qualitative explanations for the anomaly and describe a simple analytical model that captures this phenomenon succinctly. We study how various factors of real-time and TCP traffic affect the loss performance in Section IV. We summarise our conclusions and point to directions for future work in Section V.

II. The anomaly

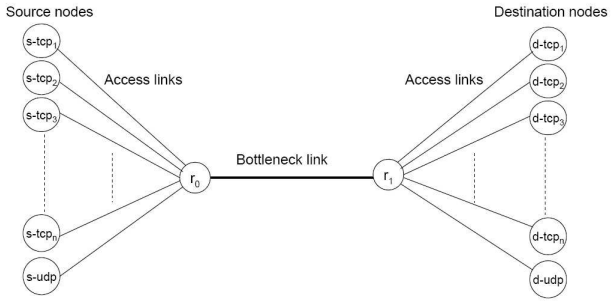


Fig. 1. ns2 simulation topology

To illustrate the anomalous loss behaviour, we require a topology that captures TCP and UDP traffic flowing through a bottleneck link router. We use $ns2$ [17] (version 2.30) simulator on the well-known dumbbell topology to simulate multiple TCP flows, shown in Fig. 1, which directly captures the bottleneck link, and is commonly used to analyse the performance of various congestion control algorithms, including TCP. It has been noted that TCP flows in *ns2* tend to synchronise when there are fewer than 500 in number [18]. Hence, we consider $n = 1000$ TCP flows, corresponding to each source-destination pair *(s-tcp$_i$,d-tcp$_i$)*, $1 \leq i \leq 1000$. We use TCP-Reno in all our simulations, consistent with the TCP version used in previous related work on buffer sizing, and employ FIFO queue with drop-tail queue management, which is commonly used in most routers today. Since synchronisation of TCP flows is an undesirable effect, it has been shown in [12] that the drop-tail queue management scheme effectively alleviates synchronisation as it drops packets arbitrarily. Thus, we employ this simple drop-tail queueing policy to avoid synchronisation issues as well.

UDP traffic is generated between nodes *(s-udp,d-udp)*. It suffices to have a single UDP flow, since open-loop traffic can be aggregated. Multiple UDP flows traversing the bottleneck link can be modelled as a single UDP flow that represents the aggregate of all individual UDP flows passing through the bottleneck link. However, we need multiple TCP flows since they each react independently to the prevailing network condition and the state of the buffers. The propagation delay on the UDP access link is fixed at 5 ms, while it is uniformly distributed between $[1, 25]$ ms on the TCP access links. The propagation delay on the bottleneck link (r_0, r_1) is 50 ms; thus round-trip times vary between 102 ms and 150 ms. All TCP sources start at random times between $[0, 10]$ s. UDP source starts at time 0 s. The simulation duration is 800 s

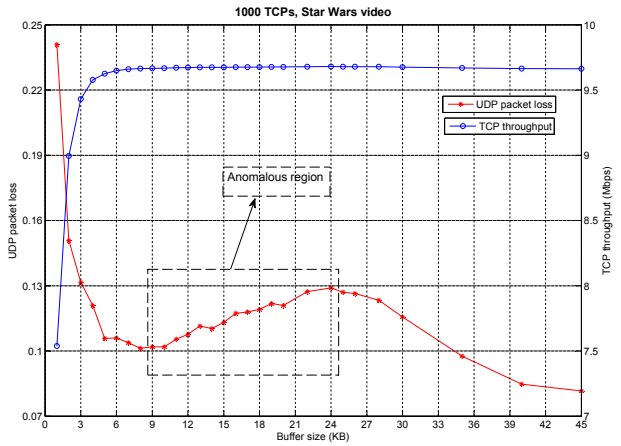


Fig. 2. Starwars video fixed packet size: UDP packet loss and TCP throughput

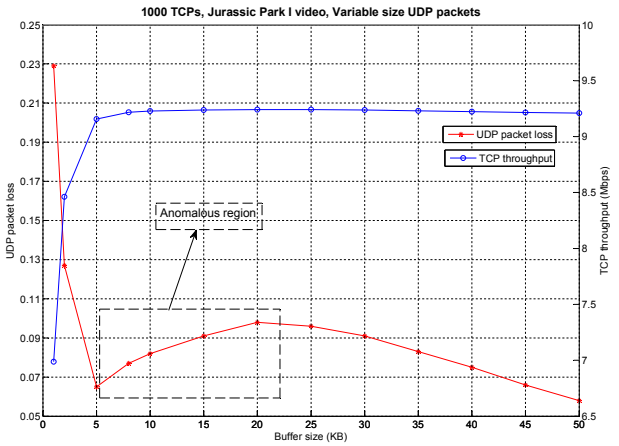


Fig. 3. Jurassic Park I video variable packet size: UDP packet loss and TCP throughput

and performance measurements are recorded after 200 s, to allow for the stabilisation of all TCP flows.

Buffer size at router r_0 is varied in terms of KiloBytes. To set the packet sizes, we draw on the fact that several real-time applications, for e.g. on-line gaming [19], use small UDP packets since they require extremely low latencies. The study showed that almost all packets were under 200 Bytes. Our experiments using Skype and Yahoo Messenger showed that for interactive voice chat, UDP packet sizes were between 150-200 Bytes. Also, traces obtained at a trans-Pacific 150 Mbps link [20] suggests that average UDP packet sizes are smaller than average TCP packet sizes. Therefore, in all our simulations, we fixed the TCP packet size at 1000 Bytes and tried fixed and variable UDP packet sizes in the range $[150, 300]$ Bytes respectively.

Akin to the traffic in the Internet core, we want to keep the fraction of UDP traffic to within 3-10% as well. We performed simulations using various movie traces such as Star Wars, Jurassic Park I, Diehard III, Silence of the lambs, Aladdin etc. For brevity, we present results from only a subset of the movies mentioned above. Results for the movies not described here closely follow the ones described. All the movie traces have been profiled and are known to exhibit self-similar and long-range-dependent traffic characteristics.

We first illustrate the phenomenon using the video traffic trace from the movie *Star Wars* obtained from [21], and references therein. The mean rate is 374.4 Kbps and the peak rate is 4.446 Mbps; the peak rate to mean rate ratio being nearly 12. The packet size is fixed at 200 Bytes. We set the bottleneck link at 10 Mbps and the TCP access links at 1 Mbps, while the UDP access link is kept at 100 Mbps. The bottleneck link was only 10 Mbps because the mean rate of the video trace (UDP) is low (374.4 Kbps), and we want to keep the fraction of UDP traffic feeding into the core to within 3-10% of the bottleneck link rate (to be consistent with the nature of Internet traffic today). In this example, the video traffic constitutes $\approx 3.75\%$ of the bottleneck link rate. Subsequent sections will present results considering higher bottleneck link rates as well.

We have a high-speed access link for UDP since UDP traffic feeding into the core can be an aggregate of many individual UDP streams. TCP traffic on the 1 Mbps access link models traffic from a typical home user. Fig. 2 shows the UDP packet loss and TCP throughput curves as a function of buffer size, along with the anomalous loss region when the buffers at router r_0 varies between 1 KB and 45 KB. We see that TCP quickly ramps up to nearly 9.6 Mbps with only about 8 KB of buffering, nearly corresponding to its saturation throughput. Simultaneously, UDP packet loss falls rapidly as well. Up to this point, both TCP and UDP behave as expected. However, the interesting phenomenon pertinent to UDP occurs around this 8 KB buffer size region. We note from the figure that increasing the buffer size from 8 KB to 24 KB actually degrades the performance of UDP traffic, i.e., UDP packet loss increases continuously as the buffer size increases within this region. The loss at 24 KB of buffering is approximately 30% more than the loss at 8 KB of buffering. There is however no appreciable increase in end-to-end TCP throughput.

To further understand the implications that variable UDP packet sizes may have on the anomaly, we performed the above simulations using video traces from the movies *Jurassic Park I* and *Diehard III* obtained from [22]. The packet sizes are uniformly distributed in the range $[150, 300]$ Bytes. All other simulation settings are identical to the above. These video traces contribute 7.7% and 7% of the bottleneck link rate respectively. Figures 3 and 4 show the corresponding UDP loss curves as a function of buffer size corresponding to the Jurassic Park I and Diehard III videos, and clearly indicate the presence of the anomaly. Simulation results with 200 Bytes fixed size packets for these videos yield a similar anomalous region, with fairly identical numbers for the measured packet loss, suggesting that the variation in the small packet sizes of UDP traffic has no significant effect on the anomaly.

We however noted only a monotonic drop in the packet loss of both TCP and UDP traffic when they existed independently of each other implying that the anomaly arises only when they coexist in the network.

Through our results in this study, we hope to bring the anomaly to the attention of network service providers who make considerable capital investment in procuring and deploying these all-optical routers, but only to obtain potentially worse performance if they inadvertently operate their buffer sizes in this anomalous region.

Having shown the phenomenon using real video traffic traces, including fixed and variable size packets, we are more interested in understanding why this counter-intuitive

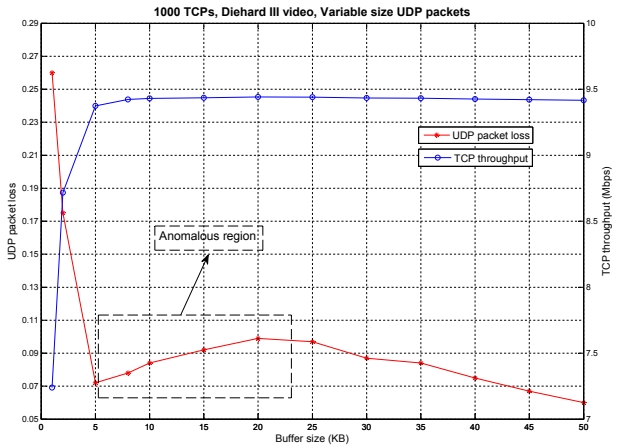


Fig. 4. Diehard III video variable packet size: UDP packet loss and TCP throughput

behaviour happens. The impact of various parameters on the UDP loss performance will be studied in Section IV. In the next section, we present an analytical model, which explains the above anomaly in detail.

III. Understanding the anomaly

We begin with an intuitive explanation of why we think the anomaly happens, and then develop a simple analytical model that explains it by quantifying the buffer sharing dynamics between TCP and real-time traffic.

When buffers at the bottleneck link are extremely small, say in the range 1-5 KB, the congestion window size for each of the TCP flows sharing the bottleneck link will also stay extremely small. TCP's congestion window is not allowed to grow beyond one or a few packets (of size 1 KB each in our simulations) in this scenario since back-to-back packets (generated by any TCP version that does not employ pacing) will be dropped at the very small buffers at the bottleneck link. The small congestion window size implies that each TCP flow transmits only a few packets in each round-trip time, and is therefore mostly idle. Consequently, the buffers at the bottleneck link are used minimally by TCP packets, and UDP has exclusive use of these buffers for the most part. This helps us understand why in this region, wherein TCP and UDP predominantly "time-share" the buffers, UDP loss decreases with buffer size, much like it would if TCP traffic were non-existent.

When buffer size is in the range 8-25 KB (corresponding to the anomaly), a larger fraction of the TCP flows are able to increase their congestion window (equivalently a smaller fraction of the TCP flows remain idle). This leads to higher usage of the buffers at the bottleneck link by TCP traffic, leaving a smaller fraction of the buffers for UDP traffic to use. The aggressive nature of TCP in increasing its congestion window to probe for additional bandwidth, causes the "space-sharing" of bottleneck-link buffers between TCP and UDP in this region to be skewed in favour of TCP, leaving lesser buffers available to UDP traffic even as buffer size increases.

We now try to quantify the above intuition via a simple analytical model that captures the transition from *time-sharing* to *space-sharing* of the bottleneck-link buffers between TCP and real-time traffic. We make the assumption that there are a sufficiently large number of TCP flows sharing the bottleneck link, and that they have sufficiently large round-trip time such that the delay-bandwidth product is larger than the buffering available at the bottleneck link. Moreover, TCP is assumed to contribute a vast majority of the overall traffic on the link (this is consistent with observations that nearly 90-95% of today's Internet traffic is carried by TCP). Under such circumstances, we first make the observation that TCP's usage of bottleneck buffers increases exponentially with the size of the buffer. More formally, let B denote the buffer size (in KB) at the bottleneck link, and $P_I(B)$ the probability that at an arbitrary instant of time the buffers at the bottleneck link are devoid of TCP traffic. Then

$$P_I(B) \approx e^{-B/B^*} \tag{1}$$

where B^* is a constant (with same unit as B) dependent on system parameters such as link capacity, number of TCP flows, round-trip times, etc. B^* can be inferred from the plot of the natural logarithm of $P_I(B)$ as a function of B, which yields a straight line. The slope of the line corresponds to $-1/B^*$.

This behaviour has been observed in the past by various researchers: by direct measurement of idle buffer probabilities [23, Sec. III], as well as indirectly via measurement of TCP throughput [4, Fig. 1]: the latter has shown roughly exponential rise in TCP throughput with bottleneck buffer size, confirming that TCP's loss in throughput (which arises from an idle buffer) falls exponentially with buffer size. We also validated this via extensive simulations (shown in Fig. 2 and in various other TCP plots in later sections) in *ns2*. 1000 TCP flows with random round-trip times from a chosen range were multiplexed at a bottleneck link, and the idle buffer probability was measured as a function of bottleneck link buffer size. The large number of flows, coupled with randomness in their round-trip times, ensures that the TCP flows do not synchronise their congestion windows. Fig. 5 plots on log-scale the idle buffer probability with bottleneck buffer size for two ranges of round-trip times, and show fairly linear behaviour in the range of 5 to 50 packets (each packet was 1 KiloByte), confirming the exponential fall as per Equation 1.

Having understood TCP's usage of the bottleneck buffers, we now consider a small fraction f (say 5 to 10%) of real-time (UDP) traffic multiplexed with the TCP traffic at the bottleneck link. The small volume of UDP traffic does not alter TCP performance significantly; however, TCP's usage of the buffer does significantly impact loss for the UDP traffic. If we assume the buffer is small (a few tens of KiloBytes), we can approximate the buffer as being in one of two states: idle (empty) or busy (full). With the objective of estimating the "effective" buffers space available to UDP traffic, we identify the following two components:

- **Fair-share:** During periods of time when TCP and UDP packets co-exist in the buffer, the buffer capacity B is shared by them in proportion to their respective rates. The first-in-first-out nature of service implies that the average time spent by a packet in the system is independent of whether the packet is UDP or TCP, and Little's law can be invoked to infer that the average number of waiting packets of a class is proportional to the arrival rate of that class. UDP packets therefore have on average access to a "fair share" of the buffers, namely fB, where f denotes the fraction of total traffic that is UDP.

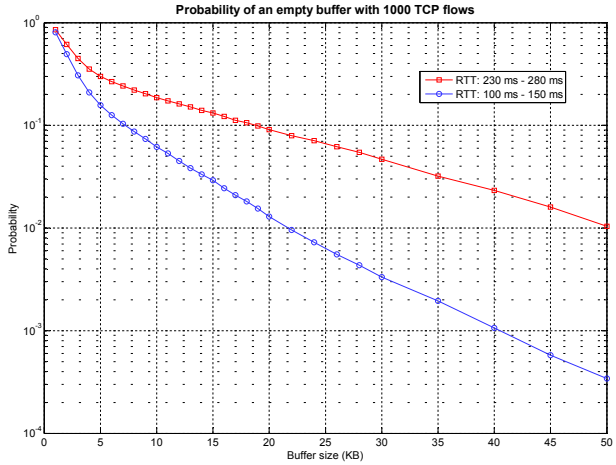


Fig. 5. Probability of idle buffer vs. buffer size for TCP traffic

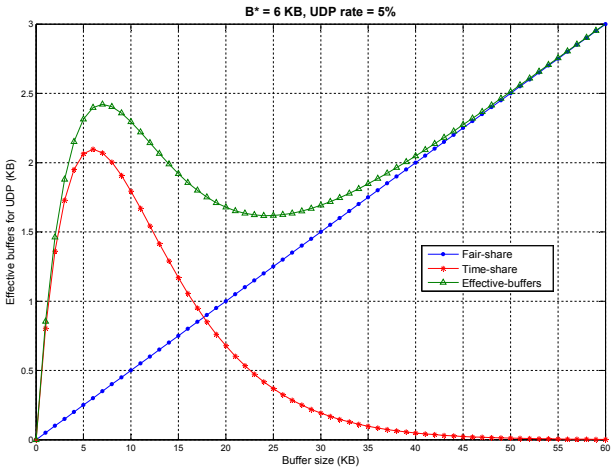


Fig. 6. Effective buffers for UDP traffic

- **Time-share:** Whenever the buffer is devoid of TCP traffic (i.e. with probability $P_I(B)$), UDP packets have access to the remaining buffer space $(1-f)B$ as well. We call this the "time share" portion, since this portion of the buffer is shared in time between UDP and TCP traffic. The time-share portion is therefore $P_I(B)(1-f)B$.

Combining the fair-share and time-share portions, and invoking Equation 1 gives us an estimate of the total "effective" buffers $\bar{B}^{udp}$ available to UDP traffic:

$$\bar{B}^{udp} = fB + (1-f)Be^{-B/B^*} \quad (2)$$

To illustrate the significance of this equation we plot it for $f = 0.05$ (i.e. 5% UDP traffic) and $B^* = 6$ KB (consistent from Fig. 5). Fig. 6 shows the total effective buffers for UDP, as well as the fair-share and time-share components. The fair-share component fB increases linearly with buffer size, while the time-share component $(1-f)Be^{-B/B^*}$ rises to a peak and then falls again (readers may notice a shape similar to the Aloha protocol's throughput curve): this happens because smaller buffers are more available for UDP to time-share, but as buffers get larger TCP permits exponentially diminishing opportunity for time-sharing. The total effective buffers for UDP, being the sum of the above two components, can therefore show anomalous behaviour, i.e., a region where larger real buffers can yield smaller effective buffers for UDP. For any realistic UDP traffic model (note that our analytical model does not make any specific assumption about the UDP traffic model), the smaller effective buffers will result in higher loss, which is of serious concern to any designer or operator of a network who operate their router buffer sizes in this region.

The model presented above is highly simplified and ignores several aspects of TCP dynamics as well as real-time traffic characteristics. It nevertheless provides valuable insight into the anomaly, and will be used in the next section for a quantitative understanding of the impact of various parameters on the severity of the anomaly.

IV. Exploring the Anomaly

Following the description of the analytical model, we are now ready to investigate the impact of various factors such as UDP traffic model, fraction of UDP traffic, number of TCP flows, round-trip times, and bottleneck link rates on the anomalous loss performance. We study the implications of these parameters in conjunction with the analytical model. All our simulations are performed for sufficiently long periods of time (400s) using *ns2* on the network topology shown in Fig. 1.

A. UDP traffic model

As noted earlier, our analytical model does not make any particular assumption about the UDP traffic model. It is therefore fairly general and predicts the inflection in effective buffer availability to UDP. We now validate that the phenomenon occurs for two different types of traffic models: short-range dependent Poisson as well as long-range dependent (LRD) Fractional Brownian Motion (fBm).

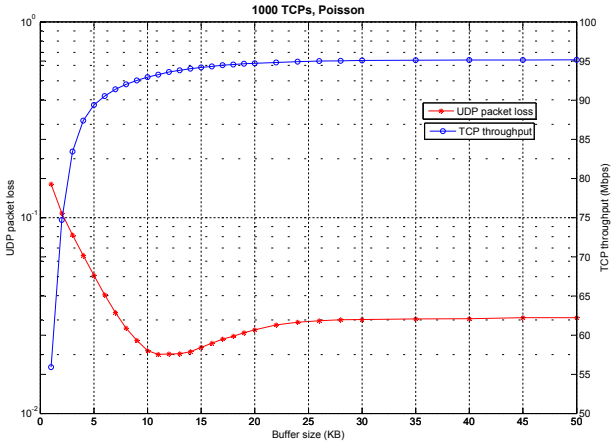


Fig. 7. Poisson model: UDP packet loss and TCP throughput

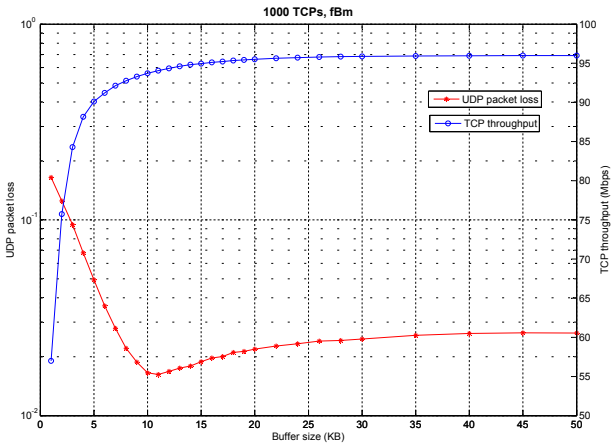
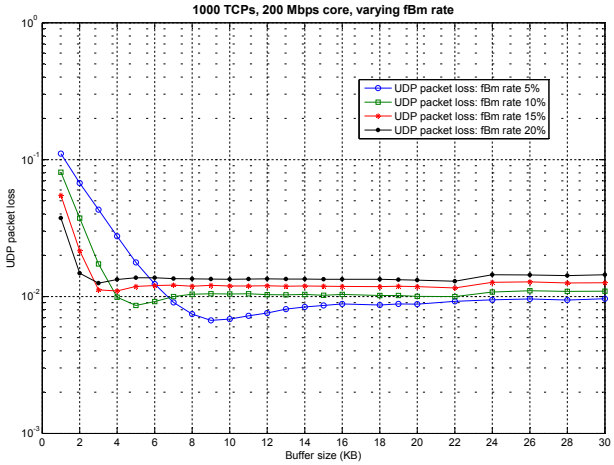
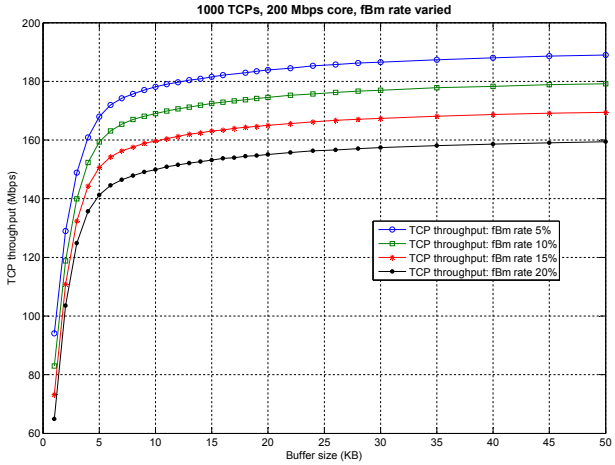


Fig. 8. fBm model: UDP packet loss and TCP throughput



(a) UDP loss with varying fBm rate



(b) TCP throughput

Fig. 9. UDP loss and TCP throughput with bottleneck link rate at 200 Mbps and fBm rate varied

1) Poisson: We start with the well-known Poisson traffic model as the UDP traffic source. The core link (we use the term core link to refer to the bottleneck link and vice-versa) bandwidth is set at 100 Mbps. TCP access links are at 10 Mbps each, while the UDP access link operates at 100 Mbps. The average rate of Poisson traffic is 5 Mbps, constituting about 5% of the total bottleneck link bandwidth. Fig. 7 shows the UDP packet loss curve (on log-scale) and the corresponding TCP throughput curve when the buffer size at router r_0 is varied from 1 KB to 50 KB. TCP is able to quickly ramp up to nearly 93 Mbps with just about 11 KB of buffering, corresponding to nearly 98% of its saturation throughput. We note from the figure that up to 11 KB, UDP packet loss falls with increasing buffer size. In addition, further increase in buffer size leads to an increase in UDP packet loss. The loss at 30 KB of buffering is 50% more than the loss at 11 KB of buffering. There is only a negligible increase in TCP throughput.

2) fBm: It is widely believed that Internet traffic is not Poisson in nature but tends to exhibit self-similar and LRD properties. To see if the phenomenon also occurs under this scenario, we generated fBm traffic at the same average rate of 5 Mbps. Other parameters are the same as before. The fBm model used is similar to our previous work in [13]. The traffic model combines a constant mean arrival rate with fractional Gaussian noise (fGn) characterised by zero mean, variance σ^2 and Hurst parameter $H \in [1/2, 1)$. We use our filtering method in [24] to generate, for a chosen H, a sequence x_i of normalised fGn (zero mean and unit variance). A discretisation interval Δt is chosen, and each x_i then denotes the amount of traffic, in addition to the constant rate stream that arrives in the i-th interval. Specifically, the traffic y_i (in bits) arriving in the i-th interval of length Δt seconds is computed using:

$$y_i = max\{0, \rho_c \Delta t + s x_i\}$$

where ρ_c denotes in bits-per-second the constant rate stream, and s is a scaling factor that determines the instantaneous burstiness. For this work we set the Hurst parameter at $H = 0.85$ and the discretisation interval $\Delta t = 1.0s$. The scaling factor s is chosen to satisfy $\rho_c \Delta t / s = 1.0$, which corresponds to moderate burstiness (around 16% of the samples are truncated), and ρ_c is then adjusted to give the desired mean traffic rate. The fluid traffic is then packetised into fixed-length packets (of size 200 Bytes) before being fed into the simulations.

We plot the UDP packet loss (on log-scale) and the TCP throughput curves as a function of buffer size in Fig. 8. Here too, as in the case of the Poisson traffic model, TCP attains 98% of its saturation throughput with only about 11 KB of buffering. UDP packet loss is the lowest at this point. An increase in buffer size negatively affects UDP packet loss, but results in only a marginal improvement in TCP throughput. The loss at 30 KB of buffering is nearly 50% more than the loss at 11 KB of buffering.

Having observed the anomalous loss phenomenon using both short-range dependent and long-range dependent traffic models, we now explore the impact of various other factors in the following subsections using the fBm traffic model (consistent with the LRD nature of Internet traffic) as the UDP source.

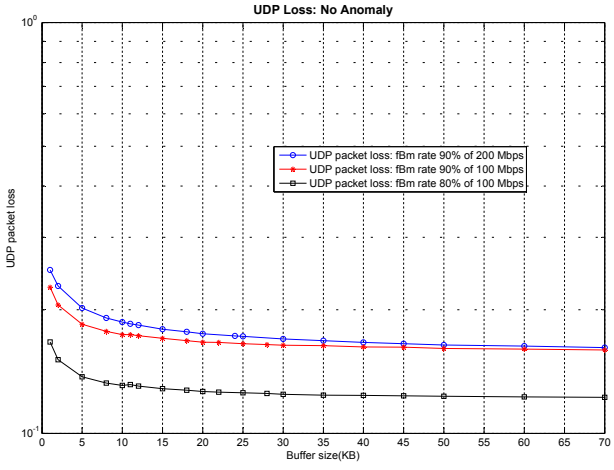
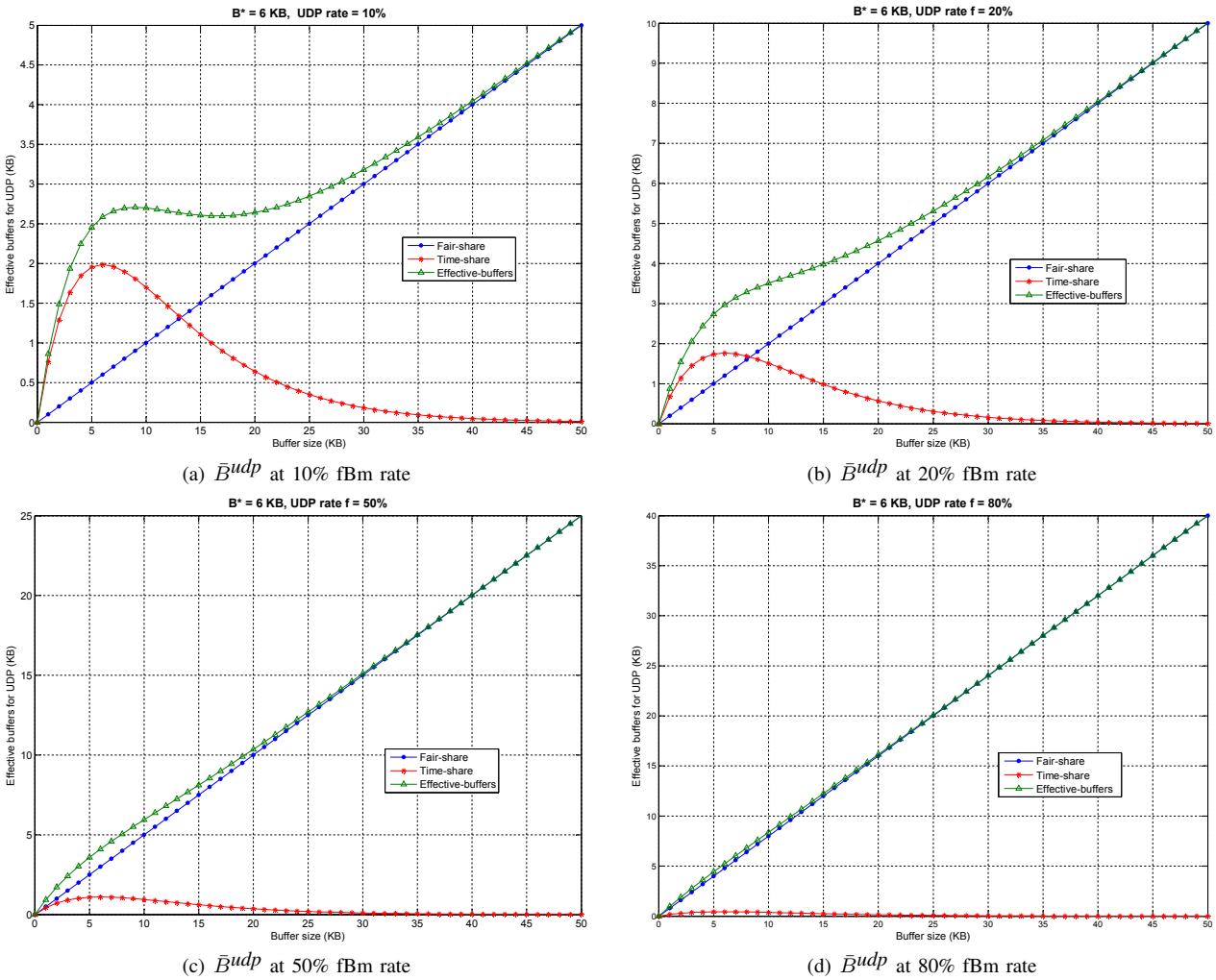


Fig. 10. UDP loss falling monotonically with buffer size by simulation

B. Fraction of UDP traffic

In this section, we are interested in answering the following question. For a fixed core link rate, will we see the inflection point if we increase the UDP rate? This is an important question to ask considering the increasing widespread use of various real-time applications in the Internet. To answer it, we simulated 1000 TCP flows on a 200 Mbps core link with the



(a) $\bar{B}^{udp}$ at 10% fBm rate

(b) $\bar{B}^{udp}$ at 20% fBm rate

(c) $\bar{B}^{udp}$ at 50% fBm rate

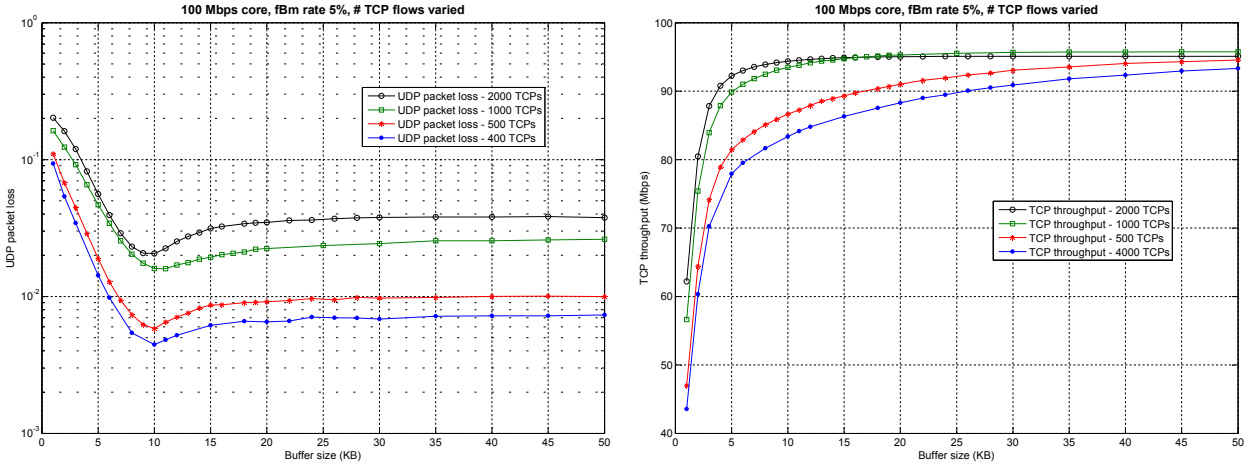
(d) $\bar{B}^{udp}$ at 80% fBm rate

Fig. 11. Effective buffers for UDP at different rates by analysis

UDP rate set at 5%, 10%, 15% and 20% of the core link rate. The resulting curves are shown in Fig. 9. We observe from Fig. 9(a) that when the UDP rate is 5%, the inflection point is clearly seen to exist at about 9 KB. Further, the inflection point gradually shifts to the left as the fraction of UDP traffic increases, suggesting that it is likely to vanish at high UDP rates. To see if this happens, we simulated three scenarios, corresponding to 80 and 90 Mbps average UDP rates on a 100 Mbps core link, and 180 Mbps average UDP rate on a 200 Mbps core link, each with 1000 TCP flows. The fraction of UDP traffic being nearly 80-90%. The resulting UDP loss curves are plotted in Fig. 10. Clearly, we can see that the UDP loss curves do not exhibit a point of inflection, i.e., there is no anomalous loss. Instead, UDP loss falls monotonically with increasing buffer size; confirming our earlier intuition.

We now provide a qualitative explanation for why the anomaly vanishes at high UDP rates. Referring back to the case when UDP rates are low, increasing buffers in the anomalous region gave TCP an exponentially larger opportunity to use the overall buffers, while giving UDP only a minimal fair-share of extra buffering; the net effect being a reduction in the effective buffers available to UDP. Now, when UDP rates are high, increasing the buffers at the bottleneck link gives UDP substantially more buffers as its fair-share (in proportion to its rate), while diminishing the opportunity for TCP to time-share the buffers with UDP. This results in a net positive gain in the effective buffers available to UDP, thereby realising monotonic packet loss with increasing buffer size. This is quantified next via our analytical model developed earlier.

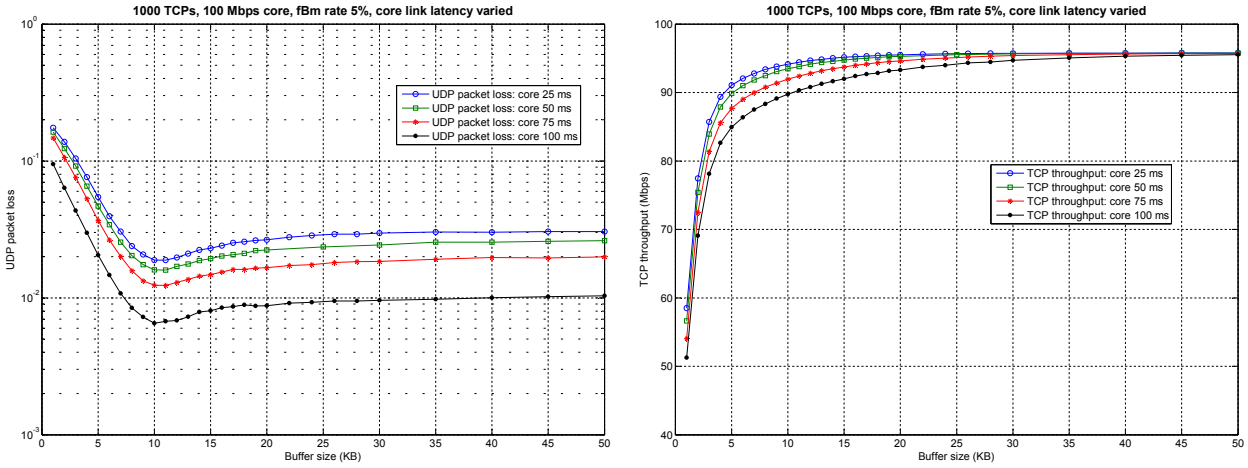
We now refer back to our analytical model (Equation 2), and draw some insights on the impact of the fair-share and time-share components on the effective-buffers available to UDP at high UDP rates. Recall that f represents the fraction of UDP traffic. We plot in Figures 11(a) – 11(d) the fair-share component, time-share component, and the effective-buffers when $B^* = 6$ KB, and the fraction of UDP traffic being 0.1, 0.2, 0.5, and 0.8, i.e., $10\%, 20\%, 50\%$, and 80% UDP traffic respectively. From the figures, and also from Fig. 6 that plots these values for $f = 0.05$ (5% UDP traffic), we note that the shape of the curves corresponding to the time-share component and the effective buffers available to UDP changes as the UDP rate increases. The presence of the time-share component is less pronounced, while the effective buffers approaches a straight line at higher rates. To explain the change in the nature of these curves we note that from Equation 2, as f increases, the fair-share component fB begins to dominate over the time-share component, since $(1-f)Be^{-B/B^*}$ becomes negligible (tends towards 0) at large f. This implies that the effect of the time-share component on the effective buffers available to UDP falls with increasing UDP rate (seen in the figures). As a result, $\bar{B}^{udp}$ increases linearly with buffer size B, which implies that the effective buffers available to UDP increases as the



(a) UDP loss with varying number of TCP flows

(b) TCP throughput

Fig. 12. UDP loss and TCP throughput when UDP rate is fixed at 5% and number of TCP flows is varied



(a) UDP loss with varying RTTs

(b) TCP throughput

Fig. 13. UDP loss and TCP throughput with varying RTTs

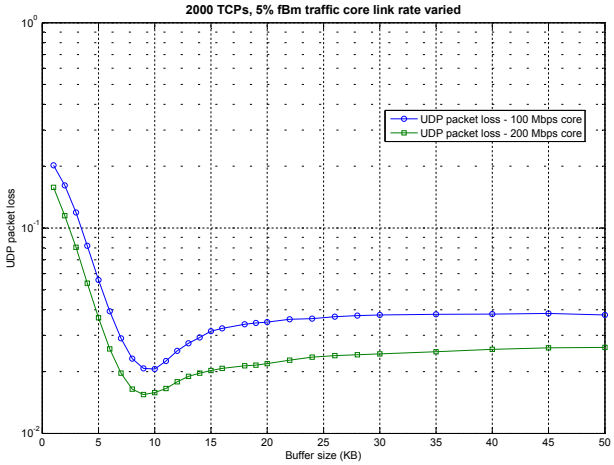
real buffer size increases, thus yielding a straight line with slope f. This explains why at high UDP rates, the packet loss curves fall monotonically with increasing buffer size.

C. Number of TCP flows and Round-trip times

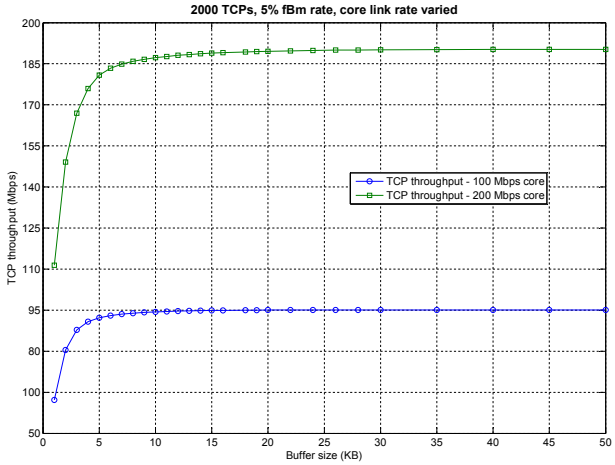
We know that tens of thousands of TCP flows traverse a core Internet router at any given time. Consequently, it is important to analyse the effect of this parameter on UDP packet loss. In this section, we first investigate the impact when the core link is 100 Mbps with 50 ms propagation delay, and the average UDP rate is about 5 Mbps (approximately 5% of the core link rate). In Fig. 12(a), we plot the UDP packet loss curves when there is network traffic from 400, 500, 1000, and 2000 TCP flows. As can be seen, the anomalous loss exists in all these scenarios, but it is interesting to note that there is very little variation in the inflection point despite the number of TCP flows increasing by a factor of five.

To understand why this is the case, we make the following observation. From Equation 1, we note that if $B = B^*$, then the probability of an empty buffer can be approximated as: $P_I(B) \approx 1/e = 0.368$, i.e., the bottleneck link is idle for $\approx 36.8\%$ of the time. This should roughly correspond to 36.8% loss in bottleneck link utilisation, or alternatively, the link is only being utilised for approximately 63.2% of the time. Since the fraction of UDP traffic is relatively small, B^* can be interpreted as the buffer size at which TCP attains $\approx 63.2\%$ of its saturation throughput. Looking closely at Fig. 12(b) we observe that for a given fraction of UDP traffic and a fixed core link rate, even as the number of TCP flows increases from 400 to 2000, the number of buffers required by TCP to attain this value does not change much; needing between 2-3 KB when there are 400-500 flows, and about 1-2 KB when there are more than 1000 flows. This suggests that the variation in B^* is not very significant, provided there exists a large number of TCP flows, which as we know is common in today's backbone routers. As a result, B^* decreases only slightly with increasing number of flows, causing only a small variation in the inflection point around the 10-11 KB buffer size value. The same argument holds if we consider core links operating at Gbps speeds. We believe that if we simulate tens of thousands of flows at Gbps core link rates (which is currently beyond the scope of the *ns2* simulator), the resulting curves will be similar to the ones shown in Fig. 12.

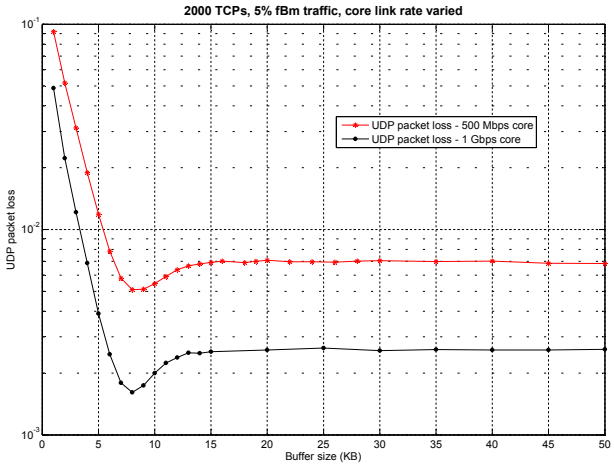
We now examine what effect round-trip times may have on the UDP loss and inflection point performance. We simulated 1000 TCP flows on a 100 Mbps core link and fixed the UDP traffic rate at 5%. Round-trip times are varied by increasing the propagation delay on the core link to successive values of 25 ms, 50 ms, 75 ms and 100 ms; thus yielding RTTs



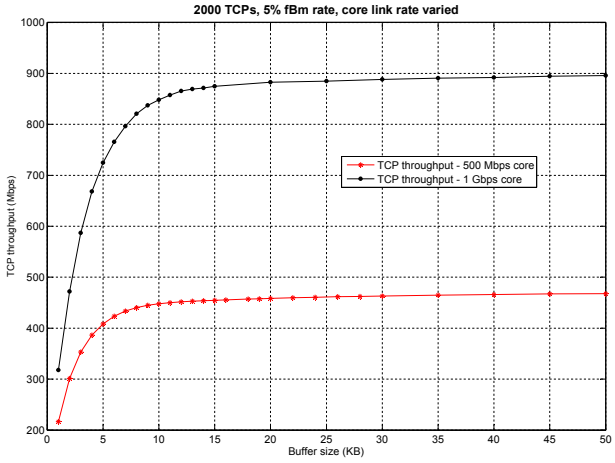
(a) UDP loss: core link rate 100 Mbps and 200 Mbps



(b) TCP throughput: core link rate 100 Mbps and 200 Mbps



(c) UDP loss: core link rate 500 Mbps and 1 Gbps



(d) TCP throughput: core link rate 500 Mbps and 1 Gbps

Fig. 14. UDP loss and TCP throughput when UDP rate is fixed at 5% and core link rate varied

in the range $[52, 100]$ ms, $[102, 150]$ ms, $[152, 200]$ ms, and $[202, 250]$ ms respectively. The resulting UDP loss and TCP throughput curves as a function of buffer size are shown in Fig. 13. From Fig. 13(a) we can observe that increasing the RTTs decreases UDP loss. Further, the inflection point does not appear to be too sensitive to varying RTTs, since there is very little movement. It corresponds to 11 KB when the RTTs are in the 52-150 ms range, and 10 KB for the 152-250 ms range. We draw upon the same argument that we used to explain why the inflection point has only a slight variation when the number of TCP flows is varied. It is easy to infer from Fig. 13(b) that for a fixed core link and UDP rate, varying the RTTs does not affect the nature of the TCP throughput curves significantly, provided there are a large number of TCP flows. The figure also suggests that for each set of RTTs, only about 2 KB of buffering suffice for TCP to attain $\approx 63.2\%$ of its saturation throughput. As a result, there is only a small variation in B^*, which suggests why the inflection point moves only slightly when the RTTs are varied.

D. Core link rates

We finally conclude our simulation study by examining the effect of core link scaling on the loss performance. Typical core link rates have grown from 100 Mbps and operate at Gbps speeds. The impact of core link scaling on TCP and UDP performance is thus another important factor to analyse as high-speed backbone links continue to evolve. Fig. 14(a) shows the UDP loss curves when the core link is set at 100 Mbps and 200 Mbps, and Fig. 14(c) shows the UDP loss curves when the core link is set at 500 Mbps and 1 Gbps. The corresponding TCP throughput curves are shown in Fig. 14(b) and Fig. 14(d). We consider 2000 TCP flows and the UDP rate is fixed at 5% of the core link rate. Round-trip time varies between $[102, 150]$ ms. The anomaly is seen to exist in each of the UDP loss curves, and we also note that there is not much variation in the inflection point with increasing core link rates.

We believe that simulating only 1000-2000 TCP flows at high core link rates (Gbps) is not very realistic; we need significantly many more TCP flows. What will be of practical interest is the dynamics of buffer occupancy when tens of thousands of TCP flows mixed with real-time traffic share a Gbps link. Unfortunately, *ns2* does not support such large scale simulation. However, based on the results, our intuition is that such a simulation will not be too different from a scenario corresponding to a 100 Mbps core link with sufficiently large number of TCP flows (1000-2000 flows). If this is the case, then the inflection point may not be very sensitive to the core link rate since there appears to be only a small variation in B^*. This implies that as the core link rates continue to scale, the increase in the amount of buffering needed may not be linear. However, this requires a much more comprehensive study.

V. Conclusions and Future Work

The study of sizing router buffers has been the subject of much attention over the past few years. Researchers have questioned the use of the rule-of-thumb and have argued that few tens of packets of buffering suffice at core Internet routers for TCP traffic to realise acceptable link utilisation. However, the research has been primarily TCP centric, since over 90% of today's Internet traffic is carried by TCP. Although real-time (UDP) traffic accounts for only about 5-10%, we note that its popularity, through the prolific use of on-line gaming, real-time video conferencing, and many other multimedia applications, is growing in the Internet. As such, we believe that the study of router buffer sizing should not focus on TCP alone, but should consider the impact of real-time traffic also.

In this paper, we examined the dynamics of UDP and TCP interaction at a bottleneck link router equipped with very small buffers. We observed a curious phenomenon - operating the buffer size in a certain region (typically between 8-25 KB) increases losses for UDP traffic as buffer size increases within this region, and results in only a marginal gain in end-to-end TCP throughput when there are a large number of TCP flows. We showed the existence of the anomalous loss behaviour using real video traffic traces, short-range dependent Poisson traffic, and long-range dependent fBm traffic models. Further, we developed a simple analytical model that gave insights into why the anomaly exists under certain circumstances. We also presented scenarios describing when the anomaly does not exist. Through extensive simulations, we investigated the impact of various factors such as fraction of UDP traffic, number of TCP flows, round-trip times, and core link rates on the anomaly. The effect of these factors on the inflection point was studied in conjunction with the analytical model. Our results inform all-optical router designers and network service providers of the presence of the anomalous region, and suggests that care must be taken when sizing all-optical router buffers in this regime since investment in larger buffers can make performance worse.

As part of our future work, we intend to conduct extensive simulations taking into account the presence of non-persistent TCP flows, i.e., TCP flows arriving and departing the network following a heavy-tailed size distribution [8]. Measurement based studies at the core of the Internet suggest that a large number of TCP flows are short-lived (non-persistent) and carry only a small volume of traffic, while a small number of TCP flows are long-lived (persistent) and carry a large volume of traffic. These flows are typically referred to as "mice" and "elephants" respectively. Given this scenario, it will be very interesting to study the interaction of UDP and TCP traffic at a bottleneck link, and in particular to see if the anomaly exists or not. Our simulation results indicate the presence of the anomaly with as few as 500-1000 persistent TCP flows. This leads us to believe that if we consider say $10,000$ TCP flows passing through the bottleneck link router, and about 10% of those to be persistent (i.e., 1000 long-lived flows), the anomaly would still occur.

We also plan to develop sophisticated analytical models [25] to further explain the anomaly, and undertake experimental study cross a trans-Australian network using the programmable NetFPGA networking hardware developed by the Stanford group [26]. Finally, we aim to perform simulations with various other versions of TCP such as TCP NewReno, BIC TCP [27], etc., and emerging congestion control algorithms designed specifically for routers with very small buffers [28].

References

[1] C. Villamizar and C. Song, "High Performance TCP In ANSNet," *ACM SIGCOMM Computer Communications Review*, vol. 24, no. 5, pp. 45-60, 1994.

[2] G. Appenzeller, I. Keslassy and N. McKeown, "Sizing Router Buffers," *Proc. ACM SIGCOMM*, Oregon, USA, Aug-Sep 2004.

[3] M. Enachescu, Y. Ganjali, A. Goel, N. McKeown and T. Roughgarden, "Part III: Routers With Very Small Buffers," *ACM SIGCOMM Computer Communications Review*, vol. 35, no. 3, pp. 83-90, Jul 2005.

[4] M. Enachescu, Y. Ganjali, A. Goel, N. McKeown and T. Roughgarden, "Routers With Very Small Buffers," *Proc. IEEE INFOCOM*, Barcelona, Spain, Apr 2006.

[5] N. Beheshti, Y. Ganjali, R. Rajaduray, D. Blumenthal and N. McKeown, "Buffer Sizing In All-Optical Packet Switches," *OFC/NFOEC*, California, USA, Mar 2006.

[6] Y. Ganjali and N. McKeown, "Update On Buffer Sizing In Internet Routers", *ACM SIGCOMM Computer Communications Review*, vol. 36, no. 5, pp. 67-70, Oct 2006.

[7] N. Hohn, D. Veitch, K. Papagiannaki and C. Diot, "Bridging Router Performance And Queuing Theory," *Proc. ACM SIGMETRICS*, New York, USA, Jun 2004.

[8] R. S. Prasad, C. Dovrolis and M. Thottan, "Router Buffer Sizing Revisited: The Role Of The Output/Input Capacity Ratio," *Proc. ACM CoNEXT*, New York, USA, Dec. 2007.

[9] G. Raina and D. Wischik, "Buffer Sizes For Large Multiplexers: TCP Queueing Theory And Instability Analysis," *Proc. EuroNGI*, Rome, Italy, Apr 2005.

[10] A. Dhamdhere and C. Dovrolis, "Open Issues In Router Buffer Sizing," *ACM SIGCOMM Computer Communications Review*, vol. 36, no. 1, pp. 87-92, Jan 2006.

[11] G.Vu-Brugier, R. S. Stanojevic, D.J.Leith and R.N.Shorten, "A Critique Of Recently Proposed Buffer-Sizing Strategies," *ACM SIGCOMM Computer Communications Review*, vol. 37, no. 1, pp. 43-47, Jan. 2007.

[12] M. Wang and Y. Ganjali, "The Effects Of Fairness In Buffer Sizing," *Proc. IFIP NETWORKING*, Atlanta, USA, May 2007.

[13] V. Sivaraman, H. ElGindy, D. Moreland and D. Ostry, "Packet Pacing In Short Buffer Optical Packet Switched Networks," *Proc. IEEE INFOCOM*, Barcelona, Spain, Apr 2006.

[14] H. Park, E. F. Burmeister, S. Bjorlin and J. E. Bowers, "40-Gb/s Optical Buffer Design and Simulations," *Proc. Numerical Simulation of Optoelectronic Devices (NUSOD)*, California, USA, Aug 2004.

[15] D. Hunter, M. Chia and I. Andonovic, "Buffering In Optical Packet Switches," *IEEE Journal of Lightwave Technology*, vol. 16, no. 12, pp. 2081-2094, Dec 1998.

[16] S. Yao, S. Dixit and B. Mukherjee, "Advances In Photonic Packet Switching: An Overview," *IEEE Communications Magazine*, vol. 38, no. 2, pp. 84-94, Feb 2000.

[17] Network Simulator ns2: www.isi.edu/nsnam/ns/

[18] G. Appenzeller, "Sizing Router Buffers," *PhD Thesis, Stanford University"*, Mar 2005.

[19] W. Feng, F. Chang, W. Feng and J. Walpole, "Provisioning On-Line Games: A Traffic analysis Of a Busy Counter-Strike Server," *ACM SIGCOMM Internet Measurement Workshop*, Nov 2002.

[20] Packet Traces From Measurement And Analysis On The WIDE Internet Backbone: http://tracer.csl.sony.co.jp/mawi/

[21] V. Markovski, F. Xue and L. Trajkovic, "Simulation And Analysis Of Packet Loss In Video Transfers Using User Datagram Protocol," *The Journal of Supercomputing*, vol. 20, no. 2, pp. 175-196, 2001.

[22] Video Traffic Traces For Performance Evaluation: http://trace.eas.asu.edu/TRACE/ltvt.html

[23] L. Andrew, T. Cui, J. Sun, M. Zukerman, K. Ho, and S. Chan, "Buffer Sizing For Nonhomogeneous TCP Sources", *IEEE Communications Letters*, vol. 9, no. 6, pp. 567-569, Jun 2005.

[24] D. Ostry, "Synthesis Of Accurate Fractional Gaussian Noise By Filtering," *IEEE Transactions on Information Theory*, vol. 52, no. 4, pp. 1609-1623, Apr 2006.

[25] A. Vishwanath, V. Sivaraman and G. N. Rouskas, "Are Bigger Optical Buffers Necessarily Better?," *Proc. IEEE INFOCOM Student Workshop*, Phoenix, Arizona, USA, Apr. 2008.

[26] NetFPGA: Programmable Networking Hardware, http://www.netfpga.org/

[27] L. Xu, K. Harfoush and I. Rhee, "Binary Increase Congestion Control (BIC) For Fast Long-Distance Networks," *Proc. IEEE INFOCOM*, Hong Kong, Mar. 2004.

[28] Y. Gu, D. Towsley, C. V. Hallot and H. Zhang, "Congestion Control For Small Buffer High Speed Networks," *Proc IEEE INFOCOM*, Alaska, USA, May. 2007.