The Impact of Emerging Streaming Media Applications on TCP/IP Performance *

Duke P. Hong†, Célio Albuquerque‡, Carlos Oliveira‡, and Tatsuya Suda‡

† Akamai Technologies
San Diego, CA
858-909-3198 (phone)
425-799-3490 (fax)
duke.hong@akamai.com

‡ University of California, Irvine
Irvine, CA 92697-3425
949-824-4105 (phone)
949-824-2886 (fax)
{celio,carlos,suda}@ics.uci.edu

***Keywords* –** TCP, congestion, fairness, ATM, ABR

Abstract

Emerging streaming media applications in the Internet primarily use UDP transport. The difficulty with supporting this type of traffic on the Internet are that they not only generate large volumes of traffic, but also they are not as responsive to network congestion as TCP-based applications. As a result, streaming media UDP traffic can cause two major maladies in the Internet: congestion collapse and unfair allocations of bandwidth among competing traffic flows. A solution to these maladies is available in many Internet environments. The Internet backbone, various ISPs, and DSL access networks rely on ATM as their layer-2 transport technology, and in such environments, ATM's Available Bit Rate (ABR) service can efficiently address these maladies. ABR is able to avoid congestion collapse and provide fair bandwidth allocations by distributing the unutilized bandwidth fairly among competing flows. This paper presents simulation results and empirical measurements that illustrate the congestion collapse and unfairness maladies, and ATM ABR's effectiveness in addressing those maladies.

*This research is supported by the National Science Foundation through grants ANI-9903427 and NCR-9628109. It has also been supported by grants from the University of California MICRO program, Hitachi America, Standard Microsystem Corp., Canon U.S.A., Novell, Tokyo Electric Power Co., Nippon Telegraph and Telephone Corp. (NTT), Nippon Steel Information and Communication Systems Inc. (ENICOM), Fujitsu, and Matsushita Electric Industrial Co., and Fundação CAPES/Brazil.

1 Introduction

The explosive growth of web traffic in the Internet is now being matched by the explosive growth of streaming media traffic, such as from RealAudio and MP3 audio streams, and RealVideo, Quick Time and Windows Media video streams. Web traffic primarily uses TCP transport. With all web traffic users using the same flow control scheme, TCP's end-to-end, adaptive flow control has allowed many users to share the available bandwidth in the Internet relatively fairly. However, streaming media traffic primarily uses UDP transport. These traffic streams not only generate large volumes of traffic, but also they are not as responsive to network congestion as TCP-based web traffic. As a result, streaming media UDP traffic can cause two major maladies in the Internet: congestion collapse, and unfair allocations of bandwidth among competing traffic flows [1]. These maladies can be significantly mitigated by using the services provided by the Asynchronous Transfer Mode (ATM) networks that comprise much of the Internet backbone. These services can provision the network resources fairly among TCP and UDP flows. This paper will describe these two maladies, unfairness and congestion collapse, in detail and illustrate the effectiveness of solutions made possible using ATM as an underlying layer-2 technology for the Internet.

The first malady—congestion collapse from undelivered packets—arises when bandwidth is continually consumed by packets that are eventually dropped downstream before reaching their ultimate destinations [2]. Network applications are now frequently written to use transport protocols, such as UDP, which are oblivious to congestion and make little or no attempt to reduce packet transmission rates when packets are discarded by the network. In fact, during periods of congestion some applications actually *increase* their transmission rates, or send multiple copies of each packet, in an effort to ensure that their applications will be less sensitive to packet losses [3]. This only worsens the congestion collapse malady by adding yet even more traffic. Unfortunately, the Internet currently has no effective way to regulate such applications.

The second malady—unfair bandwidth allocation—arises in the Internet for a variety of reasons. The primary reason is due to the presence of network applications which do not adapt to congestion. Adaptive applications (e.g., TCP-based applications) that respond to congestion by rapidly reducing their transmission rates are likely to receive unfairly small bandwidth allocations when competing with streaming media UDP applications. The Internet protocols themselves also introduce unfairness. The TCP algorithm, for instance, inherently causes each TCP flow to receive a bandwidth that is inversely proportional to its round trip time [4]. Hence, TCP connections with short round trip times may receive unfairly large allocations of network bandwidth when compared to TCP connections with longer round trip times.

One possible solution to the congestion collapse and unfairness maladies lies in the very technology that makes up much of the Internet backbone. Envisioned as the platform from which multimedia network applications could foster, Asynchronous Transfer Mode (ATM) networks provides support for a diversity of applications and services. Its multimedia service delivery architecture can ensure that network resources are fairly divided among competing TCP and UDP flows [5]. ATM networks offer four service classes. These are constant bit rate (CBR), variable bit rate (VBR), unspecified bit rate (UBR) and available bit rate (ABR). CBR and VBR services provide guaranteed quality of service for applications such as real-time video through resource reservation. They are given precedence over ABR and UBR traffic. The resources, bandwidth and buffer space, not utilized by CBR and VBR, even if reserved, may be dynamically utilized to send data traffic using the ABR and UBR services.

ABR sources utilize feedback from the network to determine when data should be transmitted. Jain *et al.* have proposed several ABR rate control algorithms that are able to prevent congestion collapse and provide global max-min fairness to competing flows [6]. In these algorithms (e.g., ERICA, ERICA+) network switches compute and enforce fair allocations of bandwidth among competing connections.

In this paper, the performance of TCP is investigated through a combination of empirical measurements and simulated networks. In particular, the maladies of and solutions to congestion collapse and unfair bandwidth allocation are investigated when TCP connections compete with multimedia UDP flows. The remainder of this paper is organized as follows. Section II illustrates the maladies of unfairness and congestion collapse in three major Internet environments. In section III, we describe the ATM ABR service and present results of several simulations using ATM ABR. The ability of ATM ABR to prevent congestion collapse and to provide fairness to competing network flows in the Internet is illustrated through simulation results. Section IV provides some concluding remarks.

2 Performance of TCP competing with streaming media flows

In this section, we examine the problems of congestion collapse and unfair bandwidth allocations when TCP and non-adaptive, streaming media flows coexist. We study how they may occur using the shared link scenario illustrated in Figure 1 via simulations [7] and empirical measurements. In this figure, there are two traffic flows. One flow is a TCP flow from S_1 to R_1, and the other is a streaming media UDP flow from S_2 to R_2. Both flows compete for access to a shared bottleneck link (N_1-N_2).

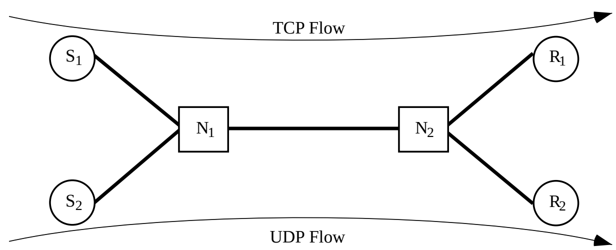

Figure 1: A network with a single shared link

Congestion collapse

The first scenario illustrates the problem of congestion collapse. In the simulations for this scenario, the TCP flow is generated by an application which always has data to send, and the streaming media flow is an unresponsive constant bit rate UDP flow. The bottleneck link in this scenario is a 1.5 Mbps T1 link between N_1 and N_2. The UDP flow traverses a second potential bottleneck link (N_2-R_2), which has a limited capacity of 512 kbps. All other links are 10 Mbps links.

Figure 2 shows simulation results of throughput measures achieved by the TCP and UDP flows as the UDP flow's input traffic load is increased. The graph is normalized with respect to T1 capacity. The total throughput delivered by the network (i.e., the sum of the throughputs of both flows) is also shown. The drop in the total throughput illustrates that severe congestion collapse occurs as the UDP flow's transmission rate increases. This is because the UDP flow fails to respond adaptively to the discarding of its packets on the second bottleneck link (N_2-R_2). Meanwhile, these packets still consume bandwidth on the N_1-N_2 link. When the UDP input traffic load increases to the T1 capacity, the TCP flow's throughput drops nearly to zero.

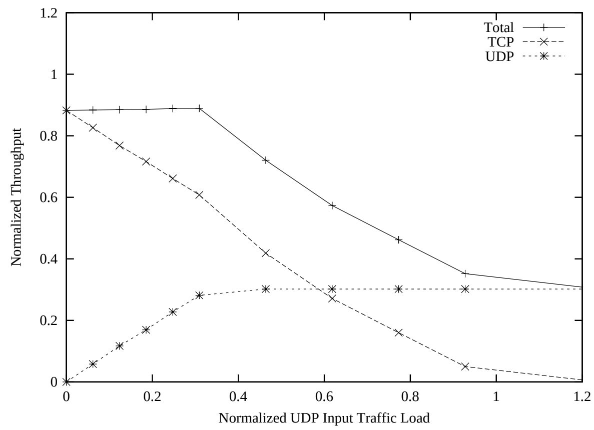

Figure 2: Congestion collapse observed as streaming media traffic load increases.

Unfairness between TCP and UDP flows

The second scenario illustrates the problem of unfair bandwidth allocations when TCP and streaming media UDP flows coexist. In this scenario, we consider the topology depicted in Figure 1 with a single bottleneck link (N_1-N_2) in three environments through empirical measures and simulations: an ATM WAN environment, a 10 base-T Ethernet LAN environment and a simulated 1.5 Mbps T1 environment. First, empirical measurements were made from a wide-area ATM network between the University of California, Irvine and the University of California, Los Angeles, over the CalRen2 network. All links including the bottleneck link (N_1-N_2) are 155 Mbps OC-3 links and are dedicated to this experiment. In this environment, the TCP flow is generated by a *ttcp* application which always has data to send, and the UDP flow is generated by an unresponsive source which transmits packets at a constant bit rate. Both applications are running on Pentium-II 400 MHz machines. Second, empirical measurements were made from a network with a 10 base-T Ethernet LAN bottleneck link (N_1-N_2), while the other links are dedicated 10 Mbps and 100 Mbps links. The TCP flow is generated by a *ftp* application transmitting a large file, and the UDP flow is generated by a set of Windows Media streams generated by a video streaming server. Third, simulations were made of a MAN with a 1.5 Mbps T1 link between (N_1-N_2). All other links are higher capacity 10 Mbps links. The TCP flow is generated by an application which always has data to send and the UDP flow is generated by an unresponsive source which transmits packets at a constant bit rate.

Figure 3 shows the TCP and UDP throughputs vs. UDP traffic generation rates, all normal-

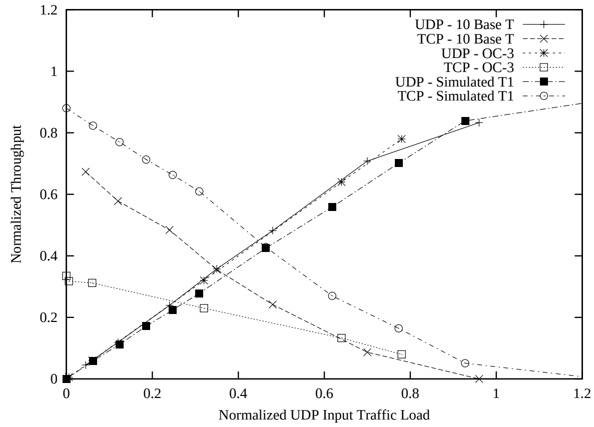

Figure 3: Unfairness as the unresponsive traffic load increases

ized with respect to the bottleneck link capacity (OC-3, 10T, or T1). Since there is only one bottleneck link (N_1-N_2) in this scenario, the max-min fair allocation of bandwidth for each of the flows is half the bottleneck link's bandwidth (if the UDP load exceeds half the bottleneck link's bandwidth). However, as Figure 3 shows, in all three environments, fairness is clearly not achieved. As the UDP traffic load increases, the throughput of TCP flows decrease down to nearly zero. This is because TCP responds to congestion while UDP does not. When the TCP flow experiences congestion, it reduces its transmission rate. This grants an unfairly large amount of bandwidth to the UDP flow. In the WAN ATM environment, the measured TCP throughput in the absence of UDP traffic is limited to approximately 50 Mbps (35% of the link capacity) by the 64 Kbyte TCP buffers and 10 ms round trip time[1]. In the Ethernet LAN environment, we measured significantly better TCP throughput, mainly because of the shorter round trip time. Collisions in the Ethernet LAN prevented TCP from achieving higher throughput. Notice that although Windows Media streams may be adaptive to congestion, they are definitely not *TCP-friendly*, and as seen in the other two environments where UDP flows are unresponsive, in this environments the UDP throughput also increases linearly with its input load. In the simulated T1 environment, TCP achieves the highest throughput, because of the low propagation delay and the absence of collisions. This figure shows that unfairness between TCP and UDP flows will occur regardless of the bottleneck link speed, network scale (LAN, MAN or WAN), and UDP

[1]The maximum TCP throughput can be estimated by the ratio of the TCP buffer size to the round trip time. 64 Kbytes/10 ms $\approx$ 50 Mbps.

application (constant bit rate or adaptive Windows Media).

Unfairness amongst TCP flows

The third experiment investigates the unfairness malady that occurs when TCP flows compete amongst themselves. In this scenario, the UDP flow is removed, and four TCP flows compete for the bottleneck link bandwidth. The T1 environment from the previous experiment is considered. Figure 4 shows the total number of bytes transmitted by each of the four TCP streams as a function of time. The figure illustrates how unfairness occurs as some TCP flows achieve higher throughput than others. The reason is that once a packet is lost for a TCP flow, it backs off, and the remaining bandwidth is rapidly used by the remaining TCP flows. This in turn makes it more difficult for a lossy TCP connection to recover its pre-packet loss transmit rate.

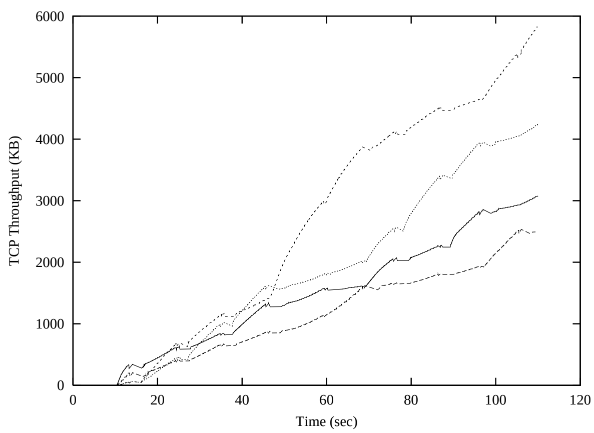

Figure 4: Unfairness amongst four competing TCP flows

Unfairness also arises as a result of the TCP algorithm inherently allocating a bandwidth which is inversely proportional to a flow's round trip time [4]. The scenario of the fourth experiment consists of a traditional parking lot topology with three flows, as shown in Figure 5. Flows B and C traverses only one bottleneck link, while flow A traverses two bottleneck links.

The experiment was run on a ATM OC-3 LAN environment and results are depicted in Figure 6. A TCP window of 64 KB was used and we varied the TCP user buffer size from 4KB to 32 KB. The results show that the TCP throughput was not sensitive to the user's buffer size. Instead the main component determining the TCP throughput was the round-trip time. Flow A experiences twice the round trip time as flows B anc C, and as consequence flow A achieves a

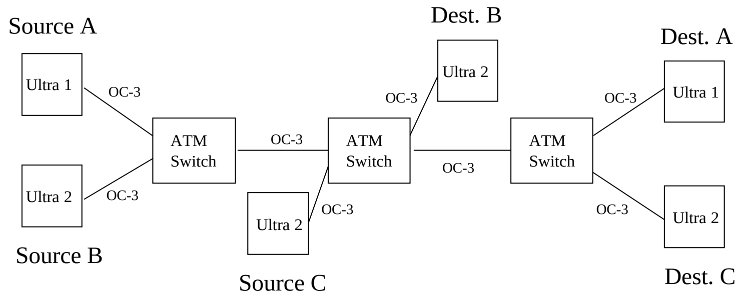

Figure 5: Parking lot configuration on an ATM OC-3 LAN environment

throughput of about 46 Mbps while flows B and C achieve a throughput of around 87 Mbps.

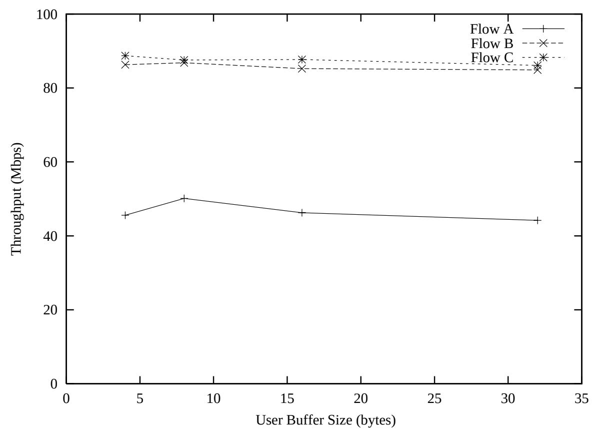

Figure 6: Unfairness amongst TCP flows in a parking lot configuration

These empirical measurements and simulation results show, that with the current Internet architecture and applications, significant congestion collapse and unfairness maladies could occur as a result of transporting streaming media over UDP.

3 Available Bit Rate Service

Many of the congestion collapse and unfairness maladies discussed earlier can be solved by effectively employing the technologies of the ATM networks that comprise much of the Internet backbone, specifically, the Available Bit Rate (ABR) service. The primary goal of the ABR service in ATM is to take advantage of bandwidth not utilized by other service classes. In the absence of native ATM applications, this purpose has been made irrelevant by streaming media applications that utilize UDP transport. While this UDP traffic and TCP-based web traffic can be delivered using ATM's UBR service, the ABR service has other features that has the potential to significantly improve network performance.

These improvements are the result of ABR's second goal. A second, equally important goal of ABR is to distribute the available bandwidth fairly among the competing ABR flows. By taking these measures, ABR is able to avoid congestion collapse and effectively address the unfairness maladies

3.1 ABR Service Description

The algorithms which implement the ABR service are described by the ATM Forum. This paper considers the ATM ABR Explicit Rate Indication for Congestion Avoidance (ERICA) algorithm described in [6].

The ERICA ABR algorithm employs a feedback loop for each ATM ABR connection (or *virtual circuit* (VC)). The feedback loop is initiated by the ATM source node by sending resource management (RM) cells for every 32 data cells sent on a VC. The RM cells are looped back to the sending ATM node upon reaching their destination ATM nodes. Congested ATM switches along the path can mark the RM cells to notify the sender that the network is congested. More importantly, switches can specify an *explicit rate* for each VC. The VCs should not exceed the explicit rates seen in their RM cells.

The ATM switches determine an explicit rate for a VC by estimating the available bandwidth for ABR sources (computed as the link capacity minus measured CBR and VBR traffic), estimating the number of active ABR sources and the bandwidth needs for each, and then allocating the available bandwidth fairly among the competing VCs. Upon seeing a VC's RM cell, in either the forward or return paths, the ATM switch may mark the RM cell with the explicit rate if it does not exceed the explicit rate already marked in the cell.

The mechanisms described above ensure two things. First, the available bandwidth is fairly divided among the competing VCs on a bottleneck link, ensuring fairness. Second, a source should not transmit at a rate higher than is possible on its most congested link even if more bandwidth is available on other links in its path, preventing congestion collapse. The following numerical results will show how these two factors will solve the congestion collapse and unfairness maladies.

3.2 Simulation Results using ATM ABR

The effectiveness of using ABR as the underlying transport mechanism for TCP and UDP in addressing the congestion collapse and unfairness maladies described earlier are illustrated below. The preceding scenarios were duplicated via simulations with ERICA ABR as the underlying ATM transport. Recommended values for ERICA sources and ATM switch parameters used in the simulations were taken from the ATM Forum standards [8]. ERICA parameters used in the simulations also included a peak cell rate for each VC equal to 10 Mbps with initial and minimum cell rates of 56 kbps. Simulation results showing the effectiveness of TCP and UDP over ATM ABR are shown in Figures 7 to 9.

Preventing congestion collapse

The congestion collapse problem occurs when the UDP streams continue to send high volumes of traffic even if much of this traffic is eventually discarded due to downstream bottlenecks. This problem was illustrated earlier in Figure 2 where the UBR flow dominated a shared T1 link (N_1 - N_2) even though the UBR flow was bottlenecked downstream by a 512 kbps link. As a result, as the figure illustrated, the throughput on the T1 link collapsed.

For the same scenarios but with ABR, Figure 7 shows the TCP, UBR and total throughput normalized with respect to the T1 capacity. The figure shows that, with ABR control, the amount of UBR traffic will not exceed the capacity of its most congested link, in this case 512 kbps, even as the UBR traffic generation rate at the source increases beyond the bottleneck link capacity. This allows the TCP stream to make use of the remaining T1 shared link. Note that the data in the figure are normalized to the T1 capacity. The total throughput over the T1 link is thus able to remain constant regardless of UDP traffic generation rate. Congestion collapse was avoided because the ERICA switch N_2 used explicit rates to limit the UDP sender S_2 to 512 kbps. As such, the remaining T1 link bandwidth (i.e., 1 Mbps) was available for the TCP flow.

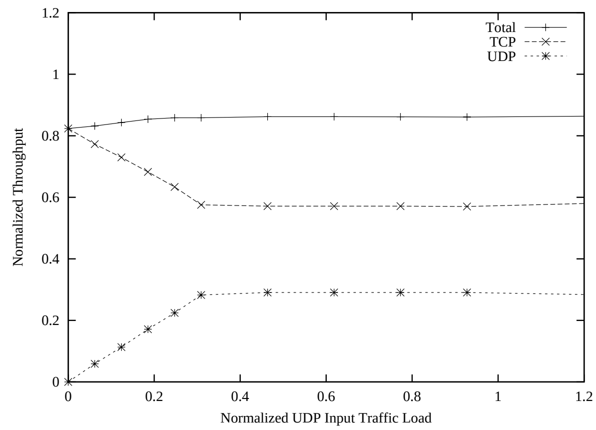

Figure 7: Simulation estimates of TCP and UDP throughput using ERICA

Providing for fairness between TCP and UDP flows

The unfairness between TCP and UDP flows are also resolved to a significant extent using ABR as the underlying transport. The unfairness maladies, illustrated earlier in Figure 3, is caused by the UDP sender continuously sending traffic even during congestion, while the TCP sender will reduce its transmit rate. ABR mitigates these maladies by allocating the available bandwidth of the congested link using explicit rate marking.

For the same scenarios but with ABR, Figure 8 shows that the ERICA ABR is able to mitigate the unfairness malady. The figure shows that as the UDP workload increases, the TCP stream is still able to achieve significant throughput. Under max-min fairness, the bandwidth should be equally divided between TCP and UDP. The results in Figure 8 does not show max-min fairness because UDP has a more aggressive transmission scheme as compared to TCP – TCP senders slow their transmission after each packet loss while UDP senders do not alter their transmission rates due to packet loss. However, while max-min fairness is not achieved, a much greater degree of fairness between the TCP and UDP flows was achieved using ERICA (Figure 8) than without it (Figure 3).

Providing for fairness amongst TCP flows

Unfairness between TCP flows in the absence of UDP flows can also be addressed using ABR. Earlier, in Figure 4, it was shown that with 4 competing TCP streams, lossy TCP connections can be "penalized" and achieve low throughput. The highest throughput stream had triple the

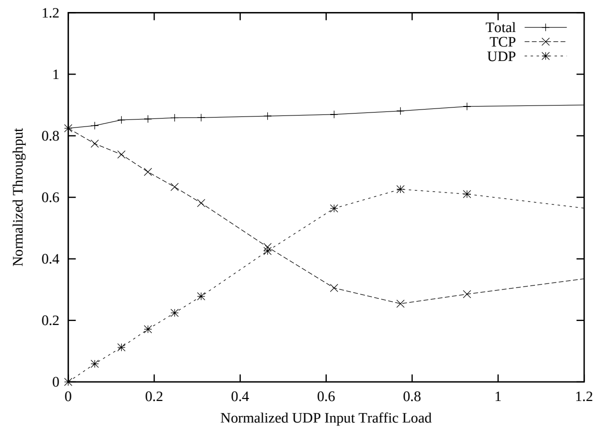

Figure 8: Simulation estimates of TCP and UDP throughput using ERICA, T1 bottleneck

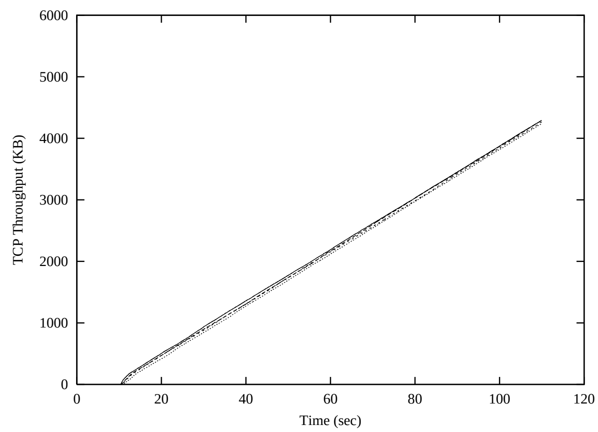

Figure 9: Fair throughput of competing TCP senders using ERICA

throughput of the the lowest throughput stream.

For the same scenarios but with ABR, Figure 9 shows that using ABR, the TCP flows, each using a separate ATM connection, are fairly allocated bandwidth. While previous results without ABR (Figure 4) showed a high degree of unfairness, the simulation results with ABR, illustrated in Figure 9, show the 4 TCP streams were able to transmit the same amount of data over the course of the entire simulation. This was due to the fact that ERICA divides the available bandwidth evenly among competing flows.

4 Conclusion

Empirical measurements and simulation results in this paper showed that TCP performance suffers from the problems of congestion collapse under heavy UDP traffic, unfairness between TCP and UDP flows, and unfairness among competing TCP flows. ATM ABR is shown to be a solution to mitigate all three of these problems and is shown to provide better performance to TCP traffic.

While ATM networks have been widely deployed, they have been mostly used as Internet Service Provider (ISP) and enterprise network backbones. ATM's services have not been fully taken advantage of. This is due in large part to the difficulty in setting up and coordinating complex services such as ABR, particularly across multiple domains. One possible migration approach is for ISPs to deploy ABR within its own domain. A recent trend has been to locate distributed web content and media delivery servers within an ISP for the users on that ISP. In this scenario, by deploying TCP over ATM ABR, an ISP would provide improved performance to its users. Simultaneously, ATM networks are emerging as a layer-2 technology for Digital Subscriber Line (DSL) access networks. In this environment, an ISP can employ ABR between the distributed content and media servers and its broadband customers or their access points, thereby mitigating congestion, providing fair bandwidth allocations and improving the performance of TCP based applications.

The unfairness between TCP and UDP flows is becoming particularly important considering the increasing use of streaming video using UDP transport over the Internet. Slow (TCP) web object retrieval times have already been observed with increasing streaming (UDP) workload. While both Windows Media and Real video players, the dominant video players used in the Internet, are beginning to employ adaptive transmission schemes, they are still likely to be much more aggressive than TCP. It is expected that congestion collapse would be avoided but the impact on fairness and perceivable TCP performance remains to be seen.

References

[1] C. Albuquerque, B.J. Vickers and T. Suda. Network Border Patrol. In *Proc. of IEEE Infocom '2000*, March 2000.

[2] S. Floyd and K. Fall. Promoting the Use of End-to-End Congestion Control in the Internet. *IEEE/ACM Transactions on Networking*, August 1999.

[3] RealNetworks Inc. Real Video Technical White Paper. *http://www.real.com/devzone/library/white–-papers/overview.html*, January 1999.

[4] J. Padhye, V. Firoiu, D. Towsley, and J. Kurose. Modeling TCP Throughput: A Simple Model and its Empirical Validation. In *Proc. of ACM SIGCOMM*, pages 303–314, September 1998.

[5] C. Tipper and J. Daigle. ATM Cell Delay and Loss for Best-Effort TCP in the Presence of Isochronous Traffic. *IEEE Journal on Selected Areas in Communications*, 13(8), October 1995.

[6] R. Jain, S. Kalyanaraman, R. Goyal, S. Fahmy, and R. Viswanathan. ERICA Switch Algorithm: A Complete Description. ATM Forum Document 96-1172, Traffic Management WG, August 1996.

[7] D. P. Hong and T. Suda. Performance of ATM Available Bit Rate for Bursty TCP Sources and Interfering Traffic. *Computer Networks and ISDN Systems*, 31, January 1999.

[8] ATM Forum. ATM Forum Traffic Management Specification Version 4.0. ATM Forum Document, Traffic Management WG, April 1996.