Enhancing the Buffer-Size for Handling Large Number of Data Packets in Routers

A. Chandrasekar#[1], M.Deepa#[2], D. Prabakar #[3], Dr. T. Kalaikumaran#[4]

[1#] *Associate Professor, Dept of CSE, SNS College of Technology-India*
E-Mail ID: chandru.as76@gmail.com
[2#] *Assistant Professor, Dept of EIE, SNS College of Technology-India*
E-Mail ID: dhipachantrasekar@gmail.com
[3#] *Assistant Professor, Dept of CSE, SNS College of Technology-India.*
E-Mail ID: prabakaralam@gmail.com
[4#] *Professor & Head, Dept of CSE, SNS College of Technology-India.*
E-Mail ID: proftkalaikumaran@gmail.com

***Abstract* - Traditionally, All Internet routers contain buffers to hold packets during times of congestion. it had been assumed that the efficiency requirements of TCP dictate that the buffer size at the router must be of the order of the bandwidth-delay (CxRTT) product. Recently, this assumption was questioned in a number of papers, and the rule was shown to be conservative for certain traffic models. Appealing to statistical multiplexing, it was shown that on a router with N long-lived connections, buffers of size O(CxRTT)/ sqrt(N) or even O(1) are sufficient. The proposed work reexamines the buffer-size requirements of core routers when flows arrive and depart. It focuses on "Can the buffers on the core routers be significantly reduced even when there are flow arrivals and departures, without compromising network performance?". If the core-to-access-speed ratio is large, then O (1) buffers are sufficient at the core routers; otherwise, larger buffer sizes are needed to improve the flow-level performance of the users. The core-to-access-speed ratio is the key parameter that determines the buffer-sizing guidelines. The proposed work also focus on heavily tailed file distribution in congested network without degrading the throughput. To develop a simple model, that provides buffer-sizing guidelines for today's high-speed routers.**

Key Words – Routers, Buffer size, Congestion, Traffic Models.

I. INTRODUCTION

Routers knit together the constituent networks of the global Internet, creating the illusion of a unified whole. While their primary role is to transfer packets from a set of input links to a set of output links, they must also deal with heterogeneous link technologies, provide scheduling support for differential service, and participate in complex distributed algorithms to generate globally coherent routing tables. These demands, along with an insatiable need for bandwidth in the Internet, complicate their design. Routers are found at every level in the Internet. Routers in access networks allow homes and small businesses to connect to an Internet Service Provider (ISP) [1][9]. Routers in enterprise networks link tens of thousands of computers within a campus or enterprise. Routers in the backbone are not usually directly accessible to end-systems. Instead, they link together ISPs and enterprise networks with long-distance trunks. The rapid growth of the Internet has created different challenges for routers in backbone, enterprise, and access networks. All Internet routers contain buffers to hold packets during times of congestion. The buffer in an Internet router has several roles. It accommodates transient bursts in traffic, without having to drop packets. It keeps a reserve of packets, so that the link doesn't go idle. It also introduces queuing delay and jitter.

Today, the buffer size in core Internet routers is typically chosen according to a rule of thumb which says: provide at least one round trip time's worth of buffering. The round trip time is often taken to be around 250ms (it takes 134ms to send a packet half way round the world and back, but queuing delay may plausibly add 100ms). A 40 GB/s line card therefore needs 1.25GByte of memory. Such large memories are hard to build in electronics. (The problem is that data can arrive at line rate, so the memory needs to be writeable at line rate. Such a high memory bandwidth is hard to engineer and DRAM access speeds increases at only 7.5% a year. Such large memories are also wildly impractical for any all-optical router that we can conceive of today. Small buffers have obvious practical benefits. In an electronic router, the buffer could be on-chip, giving much higher memory bandwidth [7]. In an all-optical router, 20-packet buffers might become feasible in the coming few years.

II. RELATED WORK

Traditionally, a buffer size of was considered necessary to maintain high utilization (here, denotes the capacity of the router and is the round-trip time) for TCP-type sources. This buffer-sizing rule implies that if there are persistent connections, each requiring a throughput of , then the buffer size should be or, in other words, the buffers should be scaled linearly with the number of flows, i.e., or . This traditional view of buffer sizing was questioned in and was shown to be outdated. By appealing to statistical multiplexing [3][11], it was shown that buffer sizes that are scaled as or are sufficient to maintain high link utilization. Another extension to the above work shows that buffer sizes can be reduced to even by smoothing the arrival process to the core. Buffer sizes can be chosen independent of the link capacity and RTT, as long as the network operator is willing to sacrifice some link utilization. In particular, it was shown that a buffer size of about 20 packets is sufficient to maintain nearly 80% link utilization (independent of the core-router capacity).

All of the above results were obtained under the assumption that there are long-lived flows in the network. The number of long-lived flows in the network was not allowed to vary with time. In reality, flows arrive and depart making the number of flows in the network random and time-varying. The main question asked is the following: "Can the buffers on the core routers be significantly reduced even when there are flow arrivals and departures, without compromising network performance". The performance metric that we use to study the impact of buffer sizing on end-user performance is the average flow completion time (AFCT) [7][2][6]. When there are file arrivals and departures, AFCT is a better metric to use than link utilization, which is the commonly used metric when there are a fixed number of flows.

Router buffers are sized today based on a rule-of-thumb commonly attributed .Using experimental measurements of at most eight TCP flows on a 40 Mb/s link, they concluded that because of the dynamics of TCP's congestion control algorithms | a router needs an amount of buffering equal to the average round-trip time of a flow that passes through the router, multiplied by the capacity of the router's network interfaces. We will show later that the rule-of-thumb does indeed make sense for one (or a small number of) long-lived TCP flows. Network operators follow the rule-of-thumb and require that router manufacturers provide 250ms (or more) of buffering. Requiring such large buffers complicates router design, and is an impediment to building routers with larger capacity.

With the continual increase in link speeds in modern communication networks, the cost of high-speed memory is likely to become a non-trivial factor in the design of data networks. For example, 100 ms of buffering in a 40 Gbs system can be quite expensive, and a typical switch is likely to have many 100 ms buffers supported on different line cards. [10][8] Therefore it is natural to ask whether the buffer need actually grow proportionally to the link speed in order to realize multiplexing gains, or whether smaller buffers will suffice. In this project we prove a strong insensitivity result that addresses this question. Our result is stronger than that in and uses a different approach.

III. MAIN IDEA

All of the above conclusions can be obtained from a single unifying model that is applicable to a large class of traffic scenarios. In particular, argue that, given a particular access- to-core-router ratio, there exists a threshold operating load below which small buffers are sufficient. Above this threshold, one would require buffers of size O(C × RTT) . A key feature of our model is that it captures the dependence between core buffer size and the number of flows in the network in figure 1. In most prior models, the number of flows and buffer size are treated as independent quantities. Previous results on networks without access speed or window size limitations indicate that small buffers are sufficient when the number of flows in the network is fixed but large. In other words, a model that assumes a fixed number of flows would indicate that buffer sizes can be reduced to 1% of the bandwidth delay product if N =10000 , independent of the maximum window size limitations and access speed limitations.

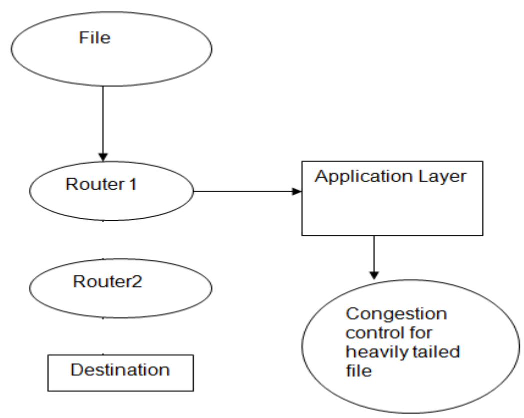

Figure 1. Work Flow diagram

A. Configuration of Networks

In this module we use to simulate TCP performance on topology with source and destination via router. The propagation delay on the access links is uniformly distributed between [1; 25]ms, while the core link (r0; r1) has a propagation delay of 50ms; the end-to-end round-trip time thus varies between 102ms and 150ms. Simulate 1000 TCP Reno flows between each source-destination pair illustrated in figure 2.Which is fairly realistic for a core link, and mitigates synchronization Issues. The buffer size at the core router r0 is varied in packets. FIFO queue with simple drop-tail queue management is employed. TCP packet size is set to 1000 Bytes.

All TCP sources start at random times between [0; 10]s. The simulation is performed for a period of 400s and only data in the interval [100; 400]s is used in the calculations.

B. Core Router with access limited networks

Will study networks where the core-router speeds are several orders of magnitude larger than the access router speeds. Before modeling arrivals and departures, first derive buffer size requirements for a network with a fixed number of long-lived flows to achieve a given link utilization. Using AFCT as the performance metric, we then consider file arrivals and departures and show that, due to access speed limitations, the core router is not congested under typical operating conditions, i.e., the number of packets dropped at the core is an order of magnitude smaller than the number dropped at the access routers at figure 3. Since the core router does not get congested, the core-router buffer size has no significant affect on the AFCT of the flows, and thus small buffers are sufficient.

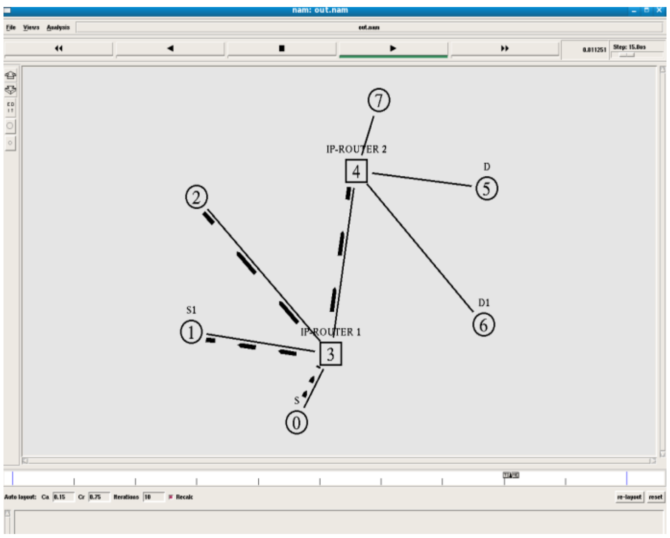

Figure 2. Packet forwarding by routers

C. Core router with unlimited access networks

Typically, there are networks where there are no access- speed limitations and each flow can potentially use a large fraction of the capacity of the link. Study the impact of small buffers in networks without access-speed limitations. it is clear that in networks with no access-speed limitations, it is impossible to reduce buffer size without seriously degrading performance. At a modest 80% load, our analysis indicates that the AFCT increases by nearly an order of magnitude when small buffers are used. Even when the load is small, say 50%, the overall AFCT doubles with the use of small buffers in the network. Thus,conclude that whenever core routers are severely congested, it is not possible to use small buffers at the routers. In fact, require O(C × RTT) buffers in order to maintain good performance to the end-users.

D .Improving throughput for heavy tailed file distribution

During the heavy load calculate the queue limit based upon the number of packet flows per second for a particular period of time. Reliable throughput is defined as the average over all file transfers (indexed by k) over time of the ratio. where Sk is the k'th file size, Wk is the total number of bytes expended to reliably transmit the k'th file, and Hk = Wk- Sk is the net overhead incurred including packet retransmissions. A file is completely determined by its size X. Each file is split into q == dX=M packets where M is the maximum segment size.

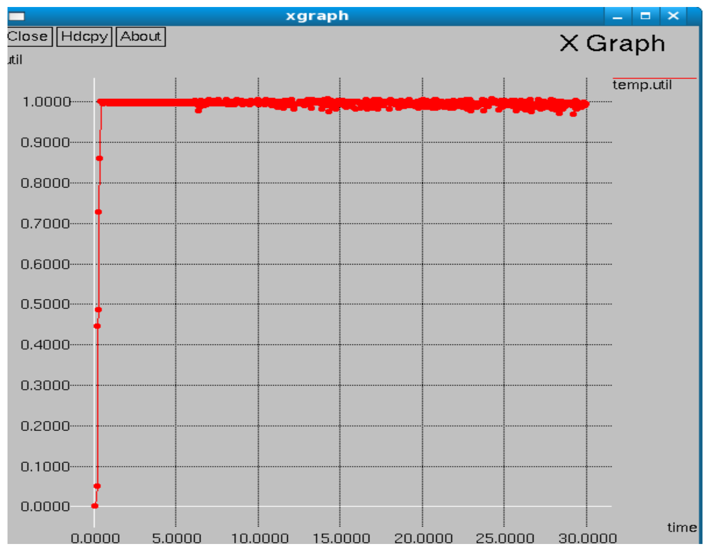

Figure 3. Packet Drop Ratio

IV. SIMULATION

NS (version 2) is an object-oriented, discrete event driven network simulator developed at UC Berkely written in C++ and OTcl. NS is primarily useful for simulating local and wide area networks. NS is an event driven network simulator developed at UC Berkeley that simulates variety of IP networks. It implements network protocols such as TCP and UPD, traffic source behavior such as FTP, Telnet, Web, CBR and VBR, router queue management mechanism such as Drop Tail, RED and CBQ, routing algorithms such as Dijkstra, and more. NS also implements multicasting and some of the MAC layer protocols for LAN simulations. To use NS, you program in OTcl script language. To setup and run a simulation network, a user should write an OTcl script that initiates an event scheduler, sets up the network topology using the network objects and the plumbing functions in the library, and tells traffic sources when to start and stop transmitting packets through the event scheduler. The term "plumbing" is used for a network setup, because setting up a network is plumbing possible data paths among network objects by setting the "neighbor" pointer of an object to the address of an appropriate object.

V. CONCLUSION

In particular, this parameter along with the buffer size determines the typical number of flows in the network. Thus the number of flows and buffer size should not be treated as independent parameters in deriving buffer-sizing guidelines and furthermore the proposed work also focus on heavily tailed file distribution in congested network without degrading the throughput. The study concludes by pointing out that it may not be possible to derive a single universal formula to dimension buffers at any router's interface in a network. we have focused on the tails of the distributions of file sizes. Distributions of file sizes observed in practice, although still heavy-tailed, are known to have a different shape for small values. Our network topology is not representative of real networks; In future this work can be implemented in different types of topology and adopted mainly for simplicity. This issue is also can being explored

ACKNOWLEDGMENT

Our Sincere thanks to our most respective Professor and Dean of the Department of Computer science and Engineering Dr.S.Karthik, Our beloved professor and Head of the Department Dr.T.KalaiKumaran and the people who are mainly motivate to prepare and publish this article in esteemed journal.
Last but not Least, We highly thanks to Dr.K.Mazhalarvizhili Madam, Professor/Head EIE to mainly suggest concluding this work.

REFERENCES

[1] Appenzeller G., Keslassy I., and McKeown N., “Sizing router buffers,” in Proc. ACM SIGCOMM, 2004, pp. 281–292.
[2] Bansal N. and Harchol-Balter M., “Analysis of SRPT scheduling: Investigating unfairness,” in Proc. ACM SIGMETRICS, 2001, pp. 279–290..
[3] Cao J. and Ramanan K., “A Poisson limit for buffer overflow probabilities,” in Proc. IEEE INFOCOM, Jun. 2002, vol. 2, pp. 994–1003.
[4] Dukkipati N. and McKeown N., “Processor sharing flows in the Internet,” High Performance Networking Group, Stanford, CA, Tech. Rep. TR04-HPNG-061604, 2004.
[5] Enachescu M., Ganjali Y., Goel A., Roughgarden T., and.McKeown N, “Part III: Routers with very small buffers,” Comput. Commun. Rev., vol. 35, no. 3, pp. 7–7, Jul. 2005.
[6] Jin C., X.Wei D., Low S. H., Buhrmaster G., Bunn J., Choe D. H., Cottrell R. L. A., Doyle J. C., Feng W.,Martin O.,Newman H., Paganini F., Ravot S., and Singh S., “Fast TCP: From theory to experiments,” IEEE Network, vol. 19, no. 1, pp. 4–11, Jan.–Feb. 2005.
[7] Kelly T.,“Scalable TCP: Improving performance in highspeed wide area networks,” Comput. Commun. Rev., vol. 33, no. 2, pp. 83–91, Apr. 2003.
[8] Lakshmikantha A., Carolyn Beck, Senior Member, IEEE, and Srikant R., Fellow, IEEE “Impact of File Arrivals and Departures on Buffer Sizing in Core Routers” April 2011
[9] Raina G., Towsley D., and Wischik D., “Part II: Control theory for buffer sizing,” Comput. Commun. Rev., vol. 35, no. 3, pp. 79–82, Jul. 2005.
[10] Wischik D. and McKeown N., “Part I: Buffer sizes for core routers,” Comput. Commun. Rev., vol. 35, no. 3, pp. 75–78, Jul. 2005.
[11] Xu L., Harfoush K., and Rhee I., “Binary increase congestion control (BIC) for fast long-distance networks,” in Proc. IEEE INFOCOM, 2004, vol. 4, pp. 2514–2524

