2012 IACSIT Hong Kong Conferences
IPCSIT vol. 30 (2012) © (2012) IACSIT Press, Singapore

Experimental Performance Study Of TCP Flows

E. Stergiou [1+], G. E. Rizos [2], D. C. Vasiliadis [2] and S.V. Margariti [1]

[1] Department of Informatics and Telecommunications Technology, GR-471 00 Arta, GREECE

[2] University of Peloponnese, Department of Computer Science and Technology, GR-221 00 Tripolis GREECE

Abstract. The flows of a Transmission Control Protocol (TCP) are an enhanced mechanism which helps to control traffic congestion more efficiently and makes the TCP connections more flexible. This study was undertaken due to an interest in having real information about the performance of TCP protocols, and in particular, to examine the role that the buffer size has with regards to intermediate routing fabrics in the TCP communication chain. This study is focused on estimating the basic link's performance metrics in terms of the utilization and throughput of TCP flows when an intermediate router uses different values of buffer size, and when various numbers of users are connected to the network, creating corresponding values of load. The results presented in this paper have been obtained by real experiment measurements, and demonstrate the TCP flow's performance, quantitatively.

Keywords: TCP protocol, throughput, experimental study, performance study, utilization of TCP, buffer size

1. Introduction

The Internet uses as a basic protocol in Transport layer (according to OSI protocol stuck) the Transmission Control Protocol (TCP), which has the ability to provide reliable communication between connected hosts [1] and [2].

A key aspect of research on the TCP protocol is an enhancing operation which uses flows to form a congestion control mechanism. In modern networks the flows stemming from a TCP protocol and congestion control techniques are very crucial for the efficient, flexible and fair utilization of resources. As well, they help avoid congestion collapse. Nevertheless, even with all of those enhancements, the TCP connection has high loss rates, especially during times of congestion. To better understand this problem we decided to study more closely the operation of a TCP protocol.

The TCP protocol implements an end-to-end window-based flow control mechanism. To detect network congestion in most of the TCP implementations, TCP sources rely mainly on packet loss, which triggers a decrease in the sender's congestion window size [3]. In recent studies it was revealed that the TCP/IP, in conjunction with end-to-end flow and congestion control, as well as intermediate nodes (e.g., routers), should control the utilization of their own resources (e.g., queues in routers) [4]. This was one of the main topics which motivated us to initiate this work, the results of which are presented here.

The other reason that led us to undertake this work came from the fact that almost all of the modern networks today use the TCP/IP protocol.

A TCP link includes at least one router in its communication chain. There are many ways in which an intermediate router can affect the TCP senders. One way is to reduce their sending rate. Another example -

+ Corresponding author. Tel.: + 30 26810 50346.
E-mail address: *ster@art.forthnet.gr.*

the most common way - is when the router drops packets when its buffer is full. This happens in cases of applying the so-called Drop-Tail routers. With Drop-Tail routers, if only one packet is lost (dropped by the router) from the sender's congestion window, it can be recovered very quickly by sending a fast retransmission and decreasing its congestion window to half its original size. However, if more than one packet is lost from the sender's congestion window, the sender will recover with timeout reducing its congestion window to one packet, and the bit rate reducing sharply the sending rate. When operating by the Drop-Tail method, losses will occur in bursts, causing timeouts.

Also, a new technique, random packet dropping strategies, has been introduced (Random Early Detection [RED] and its variants [5], [6], [7]). The RED technique is used to prevent congestion, rather than just reacting to it, by dropping packets before the router's buffers are totally exhausted. These early dropped packets are usually quickly retransmitted, therefore no timeout occurs. Both the Drop-Tail and RED-like technique tend to be against TCP flow control in router connections, thus creating unfair differences between TCP flows.

Additional techniques that are more explicit and complicated than the above listed ones have been considered (e.g., the Explicit Congestion Notification [ECN], [8]). This technique uses the TOS field in the IP header to inform senders about congestion without packet dropping. The main drawback of this method is that it requires the modification of the existing TCP/IP protocol suite.

Bearing in mind the behavior of the TCP protocol, we decided to obtain some specific measurements concerning TCP flows in order to yield real results. We decided to do that in order to get an idea of the role it plays, with regards to the buffer sizes in the intermediate network routers that interpolate in the TCP communication chain.

The remainder of this paper is organized as follows: in section II a brief analysis of TCP operation and its flows is presented. Moreover, in the same section, results of some relevant important studies are shown. Subsequently, in section III, the experimental testbed configuration is clearly explained. Section IV illustrates the results of our experiments, and comments are made with regard to the network's behavior. Finally, section V concludes with a short review and some future study directions are given.

2. Background for the TCP Operation

The most common key feature of TCP protocol operation is the window-based flow control technique. The size of a window determines the upper limit of data that can be introduced in the communication line which connects the sender and the receiver. The capacity of the connection is given by the following:

*Capacity (bits) = Bandwidth rate (bits/sec) * round-trip time (sec)*

Each TCP connection has specific data in its communication line. The data are determined by the actual size of the windows. In order to achieve its best performance, it is required that the value of the sender's data transmitted in the network be equal to the capacity of the line connection.

When the total congestion window is greater than this point, queuing at routers will increase the round-trip time (RTT) for all the connections. So, the capacity of the connection line is increased while the queue length has limited boundaries, and therefore, the capacity of connection line is also limited.

By increasing the congestion window of various connections before the capacity of the line surpasses an upper bound, the buffers start an overflowing phenomenon. So, if we are interested in avoiding buffer overflows and, hence, avoiding packet loss at the bottleneck routers, the size of the buffer must be increased. This action, in conjunction with an incremental increase in the RTT value, will result in an incremental increase in the total capacity of the lines' connection.

Today, the majority of router's fabrics have a *buffer size* of $B = C \times T$, where C depicts the capacity of the bottleneck link, and T depicts the effective two-way propagation delay of the flow that traverses the router. Appenzeller et al. [9] have suggested via their study that for large routers with lots of flows, smaller buffers can be used. They suggest *buffer size* $B = C \times T / \sqrt{N}$, where N depicts the number of long-lived TCP flows going through the router. Also, in the literature there is the Dhamdhere and Dovrolis work [10], which shows

that if the router's *buffer size* is equal to $B = C \times T$, then that leads to having higher packet loss rates in access networks, and therefore, they suggest it be used with higher values.

In addition, an alternative solution for achieving better communication performance would include the usage *buffer size* value equal to $B = O(\log W)$, as suggested by Enachescu et al. [11].

All of the studies mentioned in this section suggest that different rules lead to different *buffer sizes* when applying intermediate routers. So, the problem of intermediate router's *buffer size* is still in existence. That forces us to obtain real results from real networks to avoid various simulations' approaches that can be developed for this issue.

In the following section, the network configuration that is used for our experiments is illustrated in great detail.

3. Experimental Testbed

Because our main goal was to perform experiments under a realistic setting, we decided to use the following specific testbed configuration. We started by accepting that the TCP performance is affected not only by its protocol specification or by the algorithms that it applies. Also, we suspect that the performance of TCP is dependent on the intermediate switching system, and perhaps, also on the operating system that is used by nodes. In general, it is related by many factors of a whole network system construction.

Figure 1 shows the general diagram of an experimental network that is constructed for measuring the performance of TCP flows. One site illustrated the number of servers that are used, while another site depicted the number of clients that are connected to the network. All the servers and clients in our experiments use the Ubuntu Linux 10.4 OS. Also, each machine used in the laboratory to form the network has Intel Core 2 processors with 4 GB RAM.

A basic switch router (the Cisco 7609 Series Ethernet Services model) is used [12]. It was selected because this routing device has the ability to switch to high capacity traffic.

The bottleneck output of the intermediate router is a 1 Gigabit Ethernet link. Also, as can be seen in Figure 1, we monitor the traffic of the router's output. Moreover, the networks of both sides of the routers are also 1 Gigabit Ethernet networks.

In this specific configuration, the minimum distinct RTT values of the TCP connections fall within the time range of 30ms to 150 ms. Each client node has its own RTT value.

The traffic which is created in the servers' side is generated using an open-source Harpoon system [13], [14]. We have set the Harpoon system to create closed-loop TCP flows. The TCP traffic is generated in a random manner. As well, the load between successive transmissions is generated in random timeslots.

All the clients use Ubuntu Linux 10.4 OS and use a TCP protocol with Selective Acknowledgment as a congestion control method.

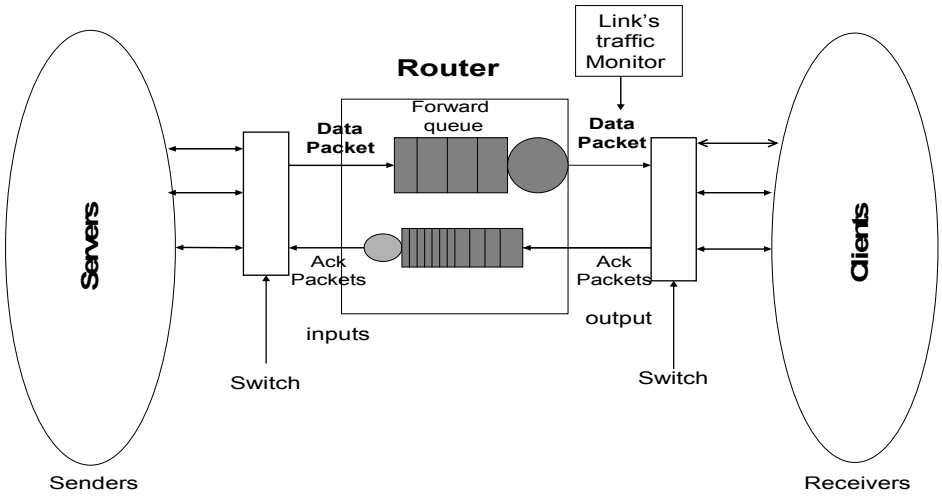

Fig. 1: Experimental testbed network

Also, the advertised TCP window size is set to 14 MB. This configuration makes the transfer to work without interrupts or unexpected delays.

In the majority of our experiments we had a normal operation of packet arrivals. That means the packet arrivals never exceeded the link's capacity. When the output link is congested,

4. Results

In the above described testbed we applied our experiments creating traffic equivalent to the load that can be created by 1000, 2000 and 3000 users, respectively. Some of the results are illustrated in the following sub-sections.

4.1. Utilization of TCP link

Figure 2 presents the *average utilization* of the router's output bottleneck link versus the router's *buffer size* for a load, which is created by $N = 10^3$, $2 \cdot 10^3$ and $3 \cdot 10^3$ users, respectively. Figure 2 depicts an average link's *utilization* of 5 minutes.

At first glance, Figure 2 seems to depict that the *utilization* of a router's output link is raised as the number of users is increased. The link's *utilization* reaches its upper bound (full utilization) with a load created by 3000 users.

However, upon looking at the values of the *buffer size*, it is obvious that for *buffer size* values greater than 400 KB, the link's *utilization* reaches saturation in almost all cases, regardless of the number of users.

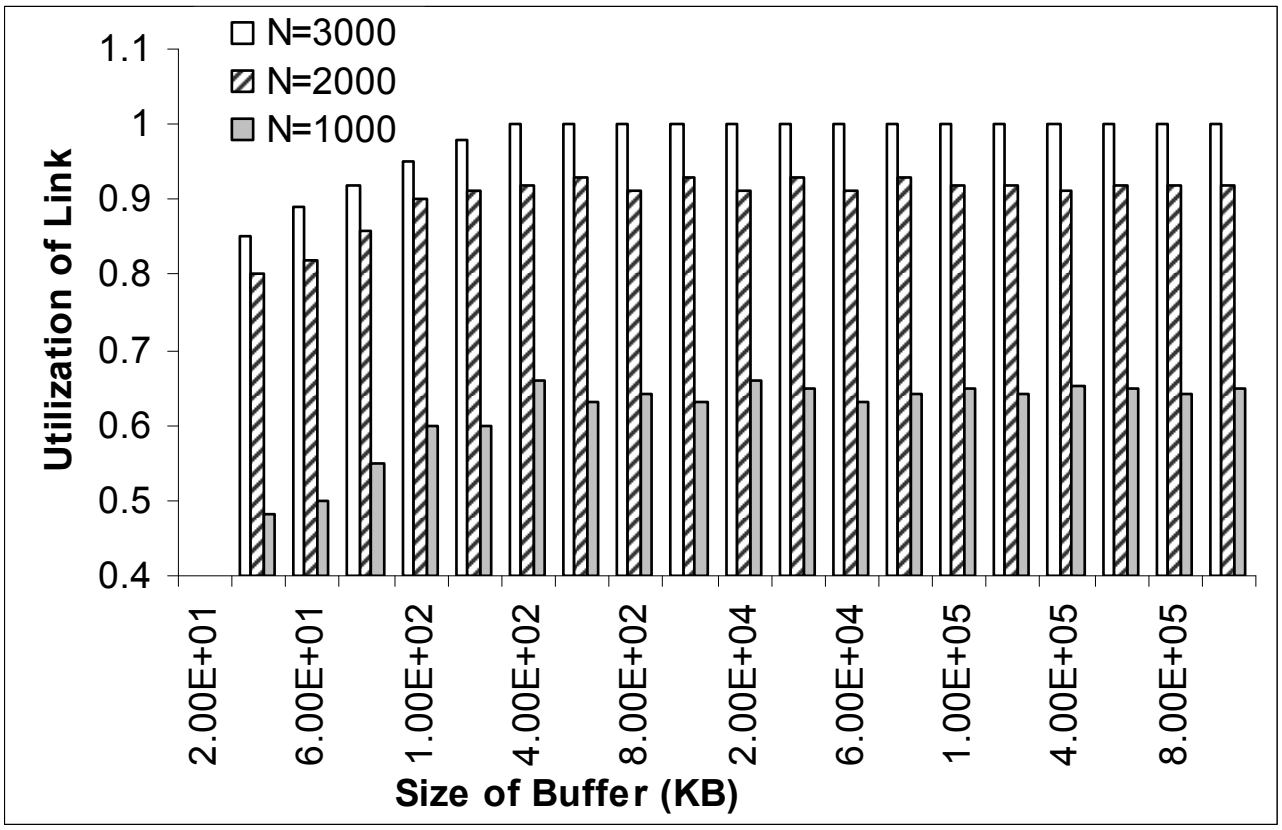

Fig. 2: Utilization of link versus buffer size of router, when the load is created by N=1000, 2000 or 3000 users, respectively

Therefore, when the intermediate router uses *buffer sizes* greater than 400 KB, we don't realize any additional advantage in the *utilization* value of the connection link.

4.2. Average throughput of TCP flow

Figure 3 depicts the *average throughput* (in kbps) of TCP flows versus a router's *buffer sizes*, for loads created by $N = 10^3$ users. In Figure 3, *average throughputs* of 100, 500 and 1000 KB TCP flows, respectively, are shown.

Figure 3 makes it apparent that the larger the TCP flow serviced by a specific network, the greater the value of *average throughput* of the TCP flow. Moreover, in a TCP flow we notice that when the router's *buffer size* exceeds a specific value, the *throughput* reaches a saturation state.

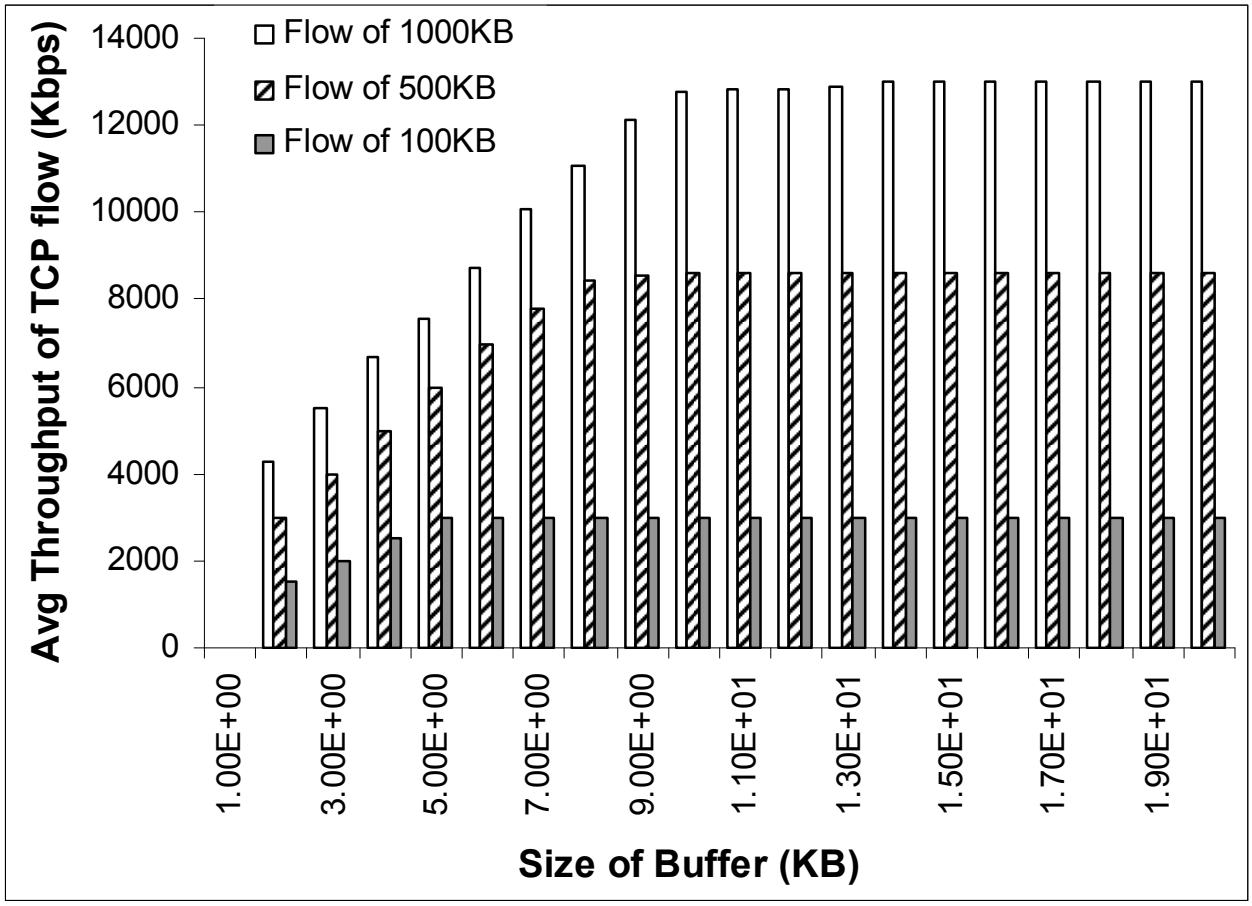

Fig. 3: Average throughput of various TCP flows versus buffer size of router, when N=1000 users are connected in the network

Figure 3 depicts that for 100 KB TCP flow, and for *buffer sizes* greater than 100 KB, the network reaches a saturation state; hence the *throughput* of flow remains stable. Similarly, for TCP flows of 500 KB and 1000 KB, respectively, the saturation point shifts in higher levels of *buffer size* values. More specifically, according to the diagram those two TCP flows start becoming saturated with *buffer size* values of 300 KB and 800 KB, respectively.

Hence, the basic result of Figure 3 is that, for a given size of TCP flows there is a bound in the value of the *buffer size*, where beyond this, the *throughput* of TCP flow remains stable.

In Figure 4, results of a slightly different scenario are shown. In that experiment a greater number of users operate on the same testbed network. In that case the performance behavior of a TCP link becomes different. In particular, Figure 4 illustrates the *average throughput* of TCP flows (in kbps) versus a router's *buffer sizes* for $N = 3 \cdot 10^3$ users. Figure 4 shows the *average throughput* for 100, 500 and 1000 KB TCP flows, respectively.

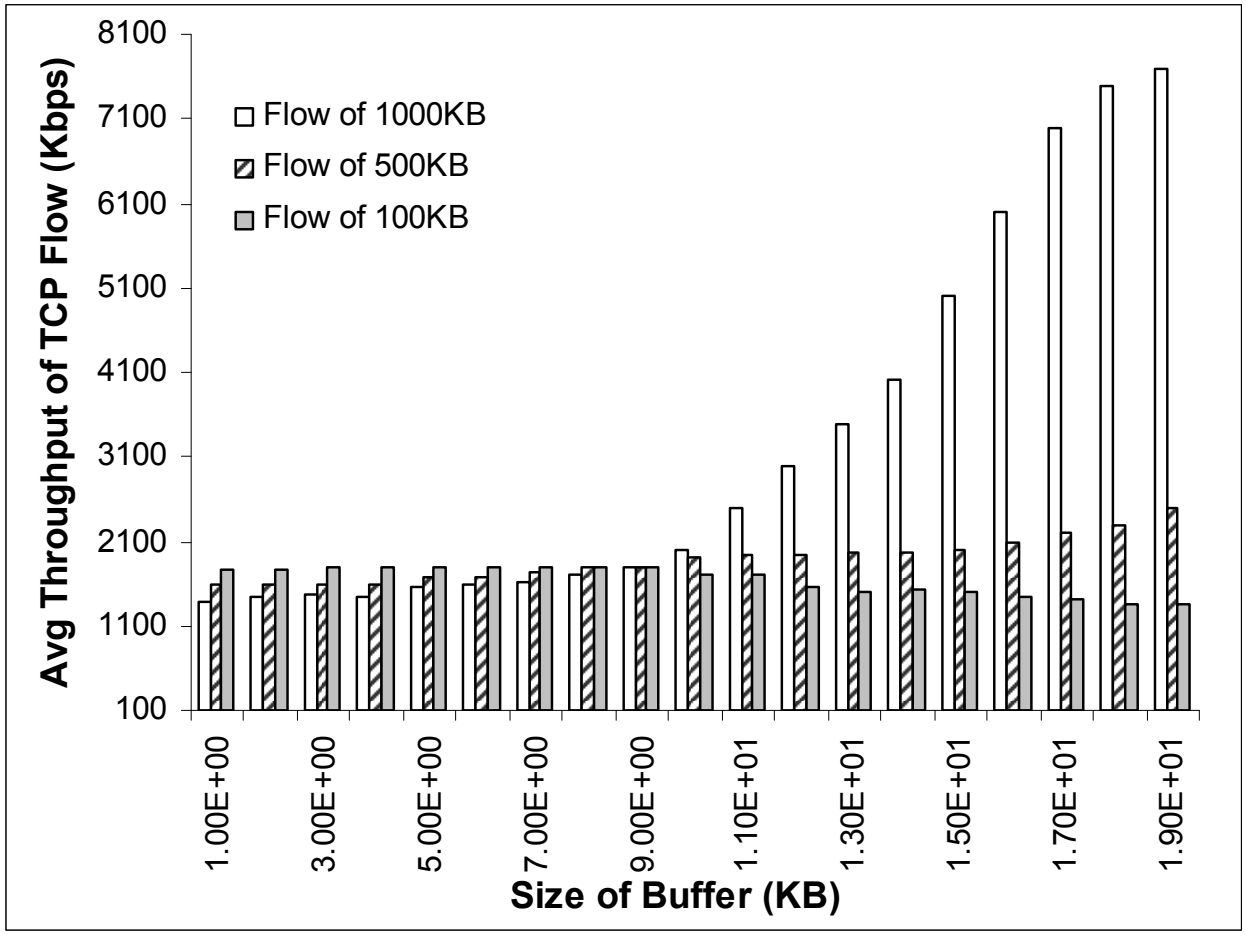

Fig. 4: Average throughput of various TCP flows versus router's buffer size, when N=3000 users are connected in the network

This diagram reveals that for *buffer size* values less than 800 KB, the *average throughput* per TCP flow remain stable and almost independent from the value of the TCP flows. Beyond the point of ~800 KB *buffer size* values, the *average throughput* of TCP flow is increased sharply as the value of the TCP flow is also increased.

In conclusion, from the results presented in Figures 3 and 4, respectively, it can be seen that there is a strong relationship between the performance of TCP flow and *buffer sizes* of an intermediate router. The high values of a router's *buffer size* don't lead to a corresponding high value in the *throughput* of TCP flows. It is always true that the optimum choice of a router's buffer is the minimum value of *buffer size* that leads to the maximum per-flow *throughput*.

4.3. Discussion

Results from the previous sub-section show that router buffer sizing considerably affects the performance of TCP flow connections. Smaller sized TCP flows require smaller buffer sized routers. This work considers closed-loop traffic or non-persistent TCP flows. However, the issue of TCP flow performance remains perplexing and significant despite all the proposals that have been introduced in the scientific literature. Our main goal here was to perform experiments under a realistic setting. To better understand buffer sizing, FPGA-based switches (which are called Controlled and Obsevable Buffer [COB] routers), and FPGA-based tools, can be employed. The FPGA-based tools are useful in traffic generation as well as in traffic measurements [14]. To integrate this study, what remains to be done are some more experiments which would exploit more complex topologies with different traffic patterns, in order to clarify the performance of a TCP communication link.

5. Conclusion

In this paper the issue of closed-loop traffic or non-persistent TCP flows is studied through experiments. This paper shows results which are realistic, as the findings were obtained from real experiments that applied non-persistent TCP flows with heavy-tailed size distributions.

In addition, benchmarking measurements, instead of obtained results from simulations, were used. The experiments revealed the relationship between the *utilization* of a bottleneck router's output link and the intermediate router's *buffer size*. The relationship between TCP flow sizes and their corresponding *throughput*, was also demonstrated. We conducted many experiments in order to estimate the average *throughput* per-flow TCP. In order to broaden this study beyond buffer sizing, the traffic pattern, network topology, remaining router settings, and the dynamics of the networks all need to be explored thoroughly. Each of the above factors should be studied in principle, individually, and then collectively.

Upon completion of the suggested future studies, some modification of the main algorithm of TCP flow control could be proposed that would avoid congestion states. We hope this work, along with others, will lead to a greater understanding of the router's buffer sizing problem and help in constructing networks with better performance.

6. References

[1] Charles M. Kozierok, The TCP/IP Guide: A Comprehensive, Illustrated Internet Protocols Reference, No Starch Press; ISBN-13: 978-1593270476, Edition October, 2005.

[2] W. Richard Stevens, TCP/IP Illustrated, Vol. 1: The Protocols, ISBN-13: 978-0201633467, Edition: Addison - Wesley Professional; January, 1994.

[3] Behrouz Forouzan, TCP/IP Protocol Suite, ISBN-13: 978-0073376042, McGraw-Hill Science/Engineering/Math; 4 Edition ; March 2009.

[4] S. Floyd, K. Fall, "Router Mechanism to Support End-to-End Congestion Control", http://www-nrg.ee.lbl.gov/floyd/papers.html

[5] S. Floyd, V. Jacobon, "Random Early Routers for Congestion Avoidance ", http://www-nrg.ee.lbl.gov/floyd/papers.html

[6] D. Lin, R. Morris, "Dynamic of Random early Detection", SIGCOMM'97.

[7] W. Feng D.D. Kandlur, D. Saha, K.G. Shin, "Techniques for Eliminating Packet Loss in Congested TCP/IP Networks", Technical Report CSE-TR-349-97, University of Michigan, Dept. of Electrical Engineering and Computer Science, Nov. 1997

[8] K.K. Remakrishnan, S. Floyd, "A Proposal to add Explicit Congestion Notification (ECN) to IP", RFC 2481, January 1999.

[9] G. Appenzeller, I. Keslassy, and N. McKeown, Sizing router buffers. In SIGCOMM '04, pages 281-292, New York, NY, USA, 2004, ACM Press.

[10] A. Dhamdhere and C. Dovrolis. Open issues in router buffer sizing. ACM Sigcomm Computer Communication Review, 36(1):87-92, January 2006.

[11] M. Enachescu, Y. Ganjali and R. Zhang-Shen . Typical versus worst case design in networking. In Proceedings of IEEE INFACOM' 06, Barcelona, Spain, April 2006.

[12] http://www.cisco.com/en/US/products/hw/routers/ps368/ps367/index.html

[13] J. Sommers and P. Barford. Self-Configuring Network Traffic Generation. In ACM/USENIX IMC, 2004.

[14] NetFPGA project http://yuba.stanford.edu/NetFPGA/