Fairness and Friendliness Analysis of Most Reliable Transport Layer Protocol

Dr. Kaushika Patel[1]

[1] Assistant Professor, Electronics Department, BVM-VVNagar, Anand
kdpatel@bvmengineering.ac.in

Abstract

Transport layer deals with process to process communication. It has reliable and non reliable services for communication. Transmission Control Protocol (TCP) is the most reliable protocol on transport layer. The basic version of TCP was designed by considering wired networks. Then other implementations could bring enhancement in basic design. The discussion is centered on one of the TCP version TCP Westwood with its New Reno implementation. Characteristics of fairness and friendliness with other competing connections have been evaluated and presented.

1 Introduction

In today's scenario Internet has become inseparable part of multimedia applications which has been resulted in the technologies that provide us the Internet access. Satellite networks are the extensive user of Internet which deals with global architecture and coverage with bandwidth adjustments. This deal with the most reliable and efficient communication protocols employed at different layers of TCP/IP protocol suit. TCP/IP protocol suit is rich in protocols employed at each layer. The model itself is not in much use but the protocols are widely used everywhere in communication field. Transmission Control Protocol (TCP) is employed in any of infrastructure base or infrastructure less networks for reliability. TCP is placed at transport layer that is implemented in different networks for process to process communication. TCP/IP is a proven best and very widely accepted suit of protocol; that has ensured quiet stable and very robust operations for any network i.e. wired/wireless.

2 Overview

TCP is enough efficient to provide the most reliable communication means for web browsing, file transfer and email. However many performances issues were there in design of conventional TCP.

The conventional TCP is engaged in any of the wired/ wireless networks, where channel noise and large latency are the major factors to degrade the performance of networks. Several surveys mention that majority of Internet traffic is taken care by TCP. TCP provides error free and low delay communication according to the assumption done based on wired networks only. The network overburden and congestion were considered to be the two reasons for data loss. Losses in TCP are detected by RTO which was triggered at initiation of connection. There is no distinction between the loss due to corruption or congestion [1].
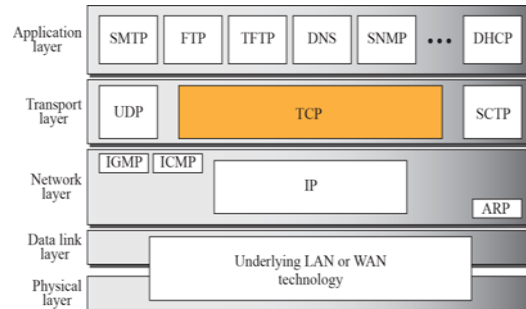
Figure 1:TCP/IP Protocol Suit

In computer networks the numbers of computers are inter connected by a common technology, they are inter connected as they are able to do the communication for exchange of information. The set of various protocols and layers is called network architecture. There are many protocols involved at each layer for proper functioning of the communication system. Each layer has predefined duties to perform and for smooth functioning of system the protocols are defined at all the layer of reference model TCP/IP, which are shown graphically in the Fig. 1. The lower layers are designed to provide services to upper layers. Upper layers are expected to get services from lower layers of protocol stack.

Majorly the work is dedicated to transport layer issues, which involves TCP, UDP and SCTP as combination of qualities of TCP and UDP.

TCP majorly provides following services. [2]
1. Stream data service
    This converts entire data in to small segments of fixed size.
2. Reliable services
    Reliability is dual of TCP when one deals with computer networks. It is the most reliable transport layer protocol as it has connection oriented and acknowledged services.
3. Full duplex service
    Two way communications with piggybacking is followed.
4. Connection oriented service
    Before data transfer, connection establishment and on completion of data transfer connection release is mandatory for TCP.

3 Congestion Control Algorithm For Various Implementations

The congestion control is heart of Transmission Control Protocol design. Algorithm for Congestion control majorly involves slow start and congestion avoidance phases. The performance of TCP for above mentioned factors could be improved by several techniques which are suggested over

the standard and basic TCP versions, they are New Reno [3] and SACK TCP [4]. Some of proactive schemes like, Westwood [7], TCP Veno [6], and TCP New Jersey [8] were effectively designed to improve data flow control and to avoid data losses by mathematically estimating certain specific network parameters. By enhancing the fundamental Tahoe TCP different renditions were developed. Tahoe TCP comprises of slow start followed by congestion avoidance and fast retransmission mechanism. TCP Reno includes "fast recovery" to the Tahoe TCP as an extra feature. TCP New Reno is a change applicable in TCP Reno, where TCP source node retransmits the segment either by receiving three dupacks or on retransmission time out. On occurrence of three dupacks the flow is decreased by reducing cwnd by half and mapped ssthresh to cwnd. If there is a loss episode detected by three dupacks, Reno enters in fast recovery and fast recovery exits with new acknowledgement. New-Reno stays in Fast Recovery until the unacknowledged data which is outstanding will get acknowledged. Selective acknowledgement is a technique which will give information about safe coming of the data by SACK option [5,6,7,8,9]. The research is oriented towards TCPW which provides end to end bandwidth estimation for utilization of available bandwidth most efficiently. So its implementation on networks results in effectiveness in resource utilization.

4 TCP-Westwood-NR [10, 11, 12]

TCP WestwoodNR is a source-side improvement for effective use of available bandwidth with New Reno employed as basic algorithm. This introduces faster recovery for avoiding the over shrinking of cwnd after reception of three duplicate ACKs. This considers the end to end estimation of bandwidth available to a TCP connection. It hugely utilizes the available resources for a particular connection and the modifications are done for improving TCP WestwoodNR. They are comparable to implementations done in the changeover from TCP Tahoe to its TCP Reno flavor. TCPWestwoodNR (TCPW) improves the performance of TCP Reno, the most widely used TCP implementation, in wired as well as wireless ad hoc networks using source side modification for TCP congestion control algorithm. The implementation is most effective for lossy links of wireless networks. Reno implementation for TCP is very sensitive to random losses whereas TCPW is not so sensitive towards random drops. Reno over reacts to the random drops, which is not a case for TCPW. The key feature of TCPW is that it doesn't require any kind of observation or interception of data packets at any proxy nodes. In place of that, TCPW is fully comprised of end to end TCP designing principle. The major ground-breaking proposal is to constantly measure TCP's sender side bandwidth used by a perticular connection through continuous monitoring rate of at which ACKs return. This estimation is then utilized to compute congestion window and slow start threshold after loss or congestion episode. This can be predicted after three duplicate acknowledgment receptions (dupacks) or after a timeout. The justification for this strategy is uncomplicated: on the contrary with TCP Reno which blindly halves the congestion window on receiving three duplicate ACKs, TCPW always attempts to draw the slow start threshold as well as congestion window that are very much consistent with effective bandwidth used. This technique of loss recovery is called faster recovery. In the basic algorithm for TCPW, 3 DUPCK always calculates slow start threshold-ssthresh = (estimated bandwidth-BWE*round trip time-RTTmin) /size of a segment-seg_size; by entering in to congestion avoidance phase. Similarly on Timeout slow start threshold-ssthresh is calculated and compared to 2, by setting it to 2 if less than 2. This set cwnd to one and enters slow start phase. Before any loss episode the TCP source will probe the network capacity. It is well understood that congestion can occur whenever very low frequency input traffic tries to exceed link capacity. The first order low pass filter is employed to obtain Low frequency components of available bandwidth for a connection. The estimation is performed by means of a low pass filter.

The section particularly includes tools and methodologies for assessment of TCPW and the improvement was actualized and compared with existing TCPW. Execution for wired, wireless networks and also remote systems were done. Essential requirement was the Linux based machine with NS-2 introduced. The experiments were done on Ubuntu 16 and ns-2.35 [13] versions. Modules of TCP Westwood and WestwoodNR are not readily available in ns-2 so they were required to install for TCP Westwood and TCP WestwoodNR [14].

5   Results

5.1   Fairness and Friendliness check with competing Reno TCP

Set of experiments were executed to check the Fairness and Friendliness property of TCPW. Here 12 nodes were created, five initial nodes n0 to n4

were assigned as source nodes. Nodes n7 to n11 were assigned to work as destinations. The intermediate nodes n5 and n6 behave as routers. All links are identified as full duplex links. The connections between n0, n1, n2, n3, n4 to n5 has link capacity of 5Mbps. Likewise the outgoing link from n6 to n7, n8, n9, n10, n11 is also defined as 5Mbps link [15]
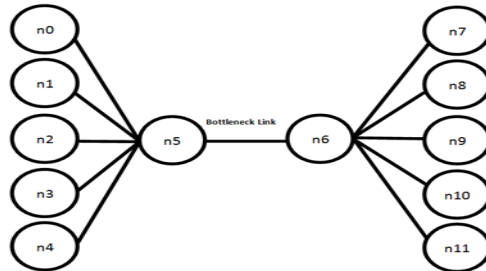

Figure 2: Congested Network for Fairness and Friendliness Check

Bottleneck was introduced between n5 to n6. The restriction was introduced as 5Mbps with 0ms. Total inflow was 25Mbps from sources to the router n5. Restriction was introduced as 5Mbps between n5 to n6 nodes. Again the flow was made relaxed between second router n6 to the destination nodes n7, n8, n9, n10, n11. Whenever congestion in the network is observed, the traditional TCP reduces the flow to half of present cwnd. TCPW calculates the ssthresh based on the link capacity. It maps cwnd to ssthresh. There may not be much difference in performance for the congested networks as traditional TCP is even conscious and sensitive towards the congestion in network. In every case flow must be reduced on observation of congestion. The drops in conventional TCPs are considered as congested network drops and have been treated like it.

5.2   Fairness and Friendliness Check with Competing Reno TCP for Error Free Link

Fairness and Friendliness are the two important properties of TCPW. The simulation experiments have been carried out for error free congested network first. In second set it was repeated for erroneous link with congestion application.
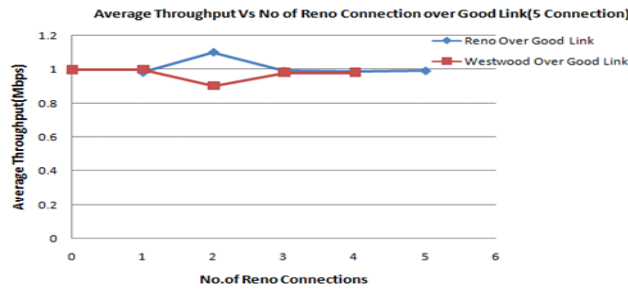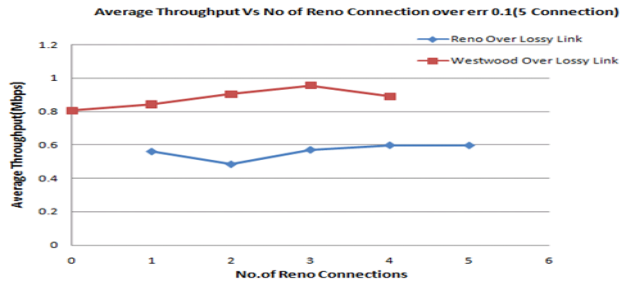
Figure 3: Fairness and Friendliness check over good link

The TCPW was evaluated for fairness and friendliness check in presence of TCP Reno over good link and lossy link (in presence of losses). The topology shown in Fig. 2 was employed with 12 nodes as discussed above. Experiments started with involving TCP Reno over all the 5 sources. The performance was analysed in presence of congestion only. From 25 Mbps to 5Mbps flow was restricted. 5Mbps link should be divided in 1Mbps for each Reno connection as fair share between those 5 connections. Link employed is lossless link so only congestion is introduced. Average of all five connections are taken and plotted in graph. Then after one of the connections was replaced by TCPW connection and four Reno connections are averaged as 1Mbps each. In next set of experiment 2 Reno connections are replaced by TCPW flows and averaged over also TCPW flows were averaged. Then 3 TCPW connections were employed and 4 and 5 connections as TCPW connections. Each time averaging was done for sharing of 5Mbps link and observations are plotted in Figure. The plots show fair sharing of the resources in absence of random drops.

5.3 Fairness and Friendliness Check with Competing Reno TCP for Erroneous Link

Another analysis was done for same scenario with random errors introduced on the bottleneck link. The error introduced was 0.1 on bottleneck of 5Mbps. When there are all TCP Reno connections involved, the average bandwidth occupied by each connection is 0.6 Mbps. Reno couldn't grab its fair share because of random drops introduced.Overall utilization of channel is 3Mbps out of 5Mbps for Reno TCP.In contradiction with this TCP WestwoodNR occupies 4Mbps by having average of 0.8Mbps each.

Figure 4: Fairness and Friendliness check over lossy link

6 Conclusion

TCPW more effectively utilizes the connection without damaging other TCPs' performance. The channel is not utilized by Reno TCP even though it is entirely available to it. Again the experiments were repeated by replacing TCP Reno by TCPW and average throughput was measured. In each case TCPW does not harm the performance of Reno TCP. It is fair and friendly to the Reno connection. TCPW utilizes the share that is not been used by other TCP connections. There is a noticeable difference in performance of both TCP implementations. This proves the fairness and friendliness property of TCPW in presence of congestion and random drops.

References

J. B. Postel, Transmission Control Protocol, IETF, RFC-793, September 1981.

Behrouz A. Forouzen, A book onTCP/IP Protocol Suite, 5th ed., McGraw-Hill Publication, 2013.

Janey C, Hoe. "Improving the Start-up Behavior of a Congestion Control Scheme for TCP", ACM Special Interest Group on Data Communication (SIGCOMM), Computer Communication Review, vol. 26, pp. 5–21, August 1996.

K. Fall, S. Floyd, Comparisons of Tahoe, Reno & SACK TCP. ftp://ftp.ee.lbl.gov/papers/sacks.ps, 1995.

Ms.Kaushika Patel, Dr.J.M.Rathod, "Comparison of TCP implementations in simulated environment" Proceedings of IEEE EESCO, pp. 1-6, January 24-25, 2015.

C. L. Zhang, C. P. Fu, Ma-Tit Yap, C. H. Foh, K. K. Wong C. T. Lau, M. K. Lai. "Dynamics Comparison of TCP Veno and Reno", Proceedings of IEEE Globecom, pp. 1329-1333, Texas USA, 2004.

C. Casetti and M. Gerla, S. Mascolo, M.Y. Sanadidi, "WestwoodNR: End-to- End Congestion Control for Wired/Wireless Networks", Wireless Networks Journal, Springer Kluwer Academic Publishers, Manufactured in The Netherlands, vol. 8, pp. 467-479, 2002.

K. Xu et al., "Improving TCP Performance in Integrated Wireless Communications Networks", Elsevier journal of Computer Networks, vol. 47, pp. 219–237, February 2005.

W. Richard Stevens. TCP Slow Start, Congestion Avoidance, Fast Retransmit, and Fast Recovery Algorithms, RFC 2001, 1997.

Mario Gerla, M. Y. Sanadidi, Ren Wang, and Andrea Zanella "*TCP Westwood: Congestion Window Control Using Bandwidth Estimation*", Proceedings of IEEE Globecom 2001 San Antonio, Texas, USA, vol.3, pp. 1698-1702, November 25-29, 2001.

. Mascolo, C. Casetti, M. Gerla, S. S. Lee, M. Sanadidi "*TCP Westwood: Congestion control with faster recovery*",UCLA Computer Science Department, Los Angeles, CSD TR200017.

M. Gerla, B.K.F. Ng, M.Y. Sanadidi, M. Valla, R. Wang "TCP westwood with adaptive bandwidth estimation to improve efficiency/friendliness tradeoffs", Journal of Elsevier Computer Communication vol.27 pp. 41-58, 2004.

ns-2, network simulator (ver.2). LBL, URL: http://www-mash.cs.berkeley.edu/ns

Kevin C. Lee, Uichin Lee, Mario Gerla, A Survey of Routing Protocols in Vehicular Ad Hoc Networks, 2006. http://www.igi-global.com/chapter/survey-routing-protocols-vehicular-hoc/43169, 2006

Ms Kaushika Patel, "Performance Enhancement Of TCP Westwood In Wired/Wireless Environment", Ph.D. Thesis submitted to CIT, Changa, 2017.