TCP Variants and Network Parameters: A Comprehensive Performance Analysis

Md. Shohidul Islam, M.A Kashem, W.H Sadid, M. A Rahman, M.N Islam, S. Anam

Abstract— **Transmission Control Protocol (TCP) includes eleven variants-Tahoe, FullTcp, TCP/Asym, Reno, Reno/Asym, Newreno, Newreno/Asym, Sack1, Fack, Vegas and VegasRBP as source and five-TCPSink, TCPSink/Asym, Sack1, DelAck and Sack1/DelAck as destination, implemented in Network Simulator (NS-2). Performance of TCP versions indicates how they respond to various network parameters-propagation delay, bandwidth, TTL (time to live), RTT (round trip time), rate of packet sending and so on. Such analysis is immensely in need to be aware of which TCP is better for a specific criterion, wherefrom an appropriate one will be selected in respective network to optimize traffic goal. But yet no complete summary is available as we've investigated hereby thoroughly. Recently published researches considered Congestion Window, Throughput, and Delay for five variants only. This paper covers all the supported variants to observe their nature regarding to five new perspectives in details.**

Index Terms— **TCP, BSS, RTT, TTL NS-2.**

I. Introduction

The reason behind the variations of TCP is that each type possesses some special criteria. Such as the base TCP has become known as TCP Tahoe. TCP Reno adds one new mechanism called Fast Recovery to TCP Tahoe [2]. TCP Newreno uses the newest retransmission mechanism of TCP Reno [8]. The use of Sacks permits the receiver to specify several additional data packets that have been received out-of-order within one dupack, instead of only the last in order packet received [7]. TCP Vegas proposes its own unique retransmission and congestion control strategies. TCP Fack is Reno TCP with forward acknowledgment [5].

Manuscript received on October 30, 2008. This work was supported in part by the IAENG Hong Kong and Department of Computer Science & Engineering, Dhaka University of Engineering & Technology, Gazipur-1700, Bangladesh.

M. Shohidul Islam, M.A. Kashem and M.N Islam are with the Department of Computer Science & Engineering DUET, Gazipur-1700, Bangladesh. (e-mail: shohidulcse@duet.ac.bd, drkashemll@duet.ac.bd, nazrul_ruet@yahoo.com). W.H Sadid is with the Concordia University, Canada. (email:whsadid@yahoo.com). M. A Rahman is with Curtin University of Technology, Australia. (email: M.Rahman@cbs.curtin.edu.au). S. Anam is B.Sc Engr. in CSE. (email: ratna_3001@yahoo.com).

II. Performance Evaluation

A. *Packet Vs Bandwidth*

Channel bandwidth has been varied each time and the number of packets was counted at destination during entire simulation period which forms fig 1.

Table 1. Packet received by TCPs for various bandwidth

TCP variants	Packets at Bandwidth			
	2 Mb	8 Mb	16 Mb	32 Mb
Tahoe	3214	3270	3280	3310
Reno	3214	3270	3280	3310
NewReno	3214	3270	3280	3310
NewrenoASYM	3247	3305	3317	3347
sack1	3214	3270	3280	3310
Vegas	3214	3270	3280	3310
Fack	3127	3191	3203	3233

As bandwidth of channel increases, number of packet received also increases by different magnitude for different variants. Highest number of packet is obtained for NewrenoASYM (Newreno in the asymmetry channel) because of its 'Fast Recovery' and 'Open Loop' congestion control mechanism. But TCP Tahoe, Reno, NewReno, sack1 and Vegas behaves identically with bandwidth variation which generate a common curve.

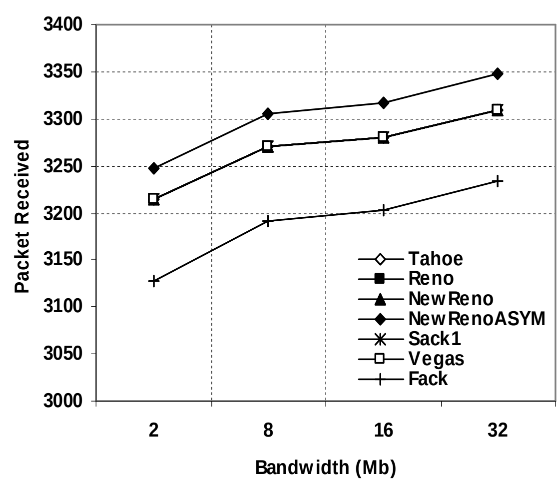

Fig. 1 Packet vs Bandwidth.

ISBN: 978-988-17012-2-0

B. Packet Vs Delay

Although number of packet received is inversely proportional to propagation delay, 'Vegas' has the best performance as depicted in fig. 2 due to its improved retransmission technique.

Table 2. Packet received by TCPs over propagation delays

TCP variants	Packets at Delay				
	10 ms	20 ms	30 ms	40 ms	50 ms
Tahoe	6869	6479	6215	5973	5673
Reno	6869	6479	6215	5973	5673
NewReno	6869	6479	6215	5973	5673
NewrenoASYM	6869	6479	6215	5973	5673
sack1	6869	6479	6215	5973	5673
Vegas	7200	6812	6400	6091	5934
Fack	6869	6479	6215	5973	5673

In Vegas, 'acknowledgment' is merged with 'data packet' (which is called *piggybacking*) instead of separate transmission .It saves fifty percent (50%) time than normal TCP implementation since 'acknowledgement' passing does not take extra time for it. That is why it can transmit more data.

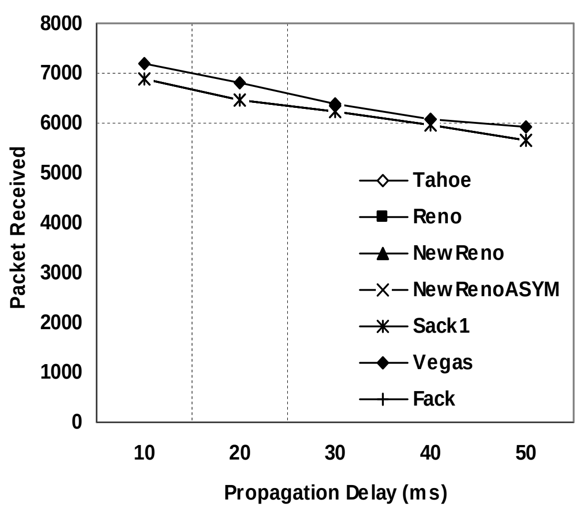

Fig. 2 Packet vs propagation delay.

C. Packet Vs Rate

The nature of TCPs to packet sending rate consists of basically two phases (fig. 3).

Linearly Increasing Phase: TCP agent increases the packet estimation to be sent after getting the acknowledgement of previously sent packets. The slope of curve at any point within this region is bounded by the range (0, 1].

Saturation Phase: TCP reaches in this phase when each agent consume their maximum allowable channel proportion which is also called **DC** state having constant slope equivalent to *zero.*

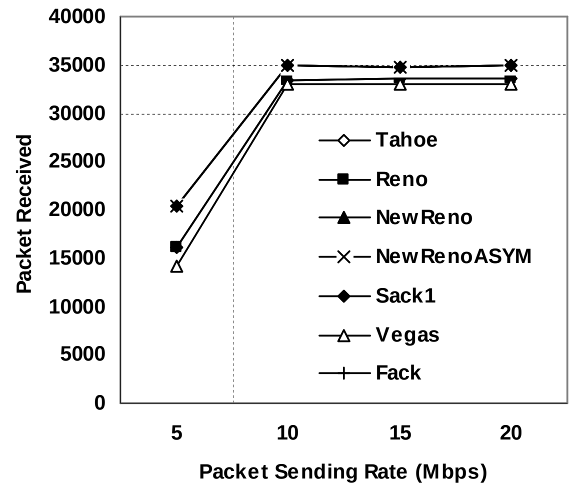

Fig. 3 Graph of packet vs rate

In this case, Newreno, NewrenoASYM, Sack1 and Fack show the best performance as they increase congestion window size exponentially with rate provided that no congestion occurred in the channel.

Table 3. Packet received by TCPs due to various transmission rates

TCP variants	Packets at Rate			
	5 Mbps	10 Mbps	15 Mbps	20 Mbps
Tahoe	16110	33388	33625	33507
Reno	16110	33388	33625	33507
Newreno	20372	34928	34809	34928
NewrenoASYM	20372	34928	34809	34928
Sack1	20372	34928	34809	34928
Vegas	14131	33009	33009	33009
Fack	20372	34928	34809	34928

D. Buffer Volume Vs Time

Buffer volume posses a close connection with 'bandwidth delay product' which is defined by- $bw\_del\_prod\,(packets) = \frac{bw\,(bits/\sec) * RTT\,(\sec)}{8 * MSS\,(bytes)}$. TCP can only achieve its optimal throughput if the minimal buffer size on the path is equal to the bandwidth delay product. If the minimal buffer size is larger, a constant backlog of packets builds up in the buffer wasting network resources and introducing latency. If it is smaller, the congestion window will never be able to reach the necessary size to utilize the available bandwidth. If the buffer size is larger than the bandwidth during *Slow Start*, the congestion window needs to be able to reach $2*(bw\_del\_prod + buffer\_size)$ to achieve optimal throughput. The receiver's advertised window, which is always an upper bound to the congestion window, has to be set to a large enough value for this growth. Window scaling might be necessary to achieve this.

ISBN: 978-988-17012-2-0

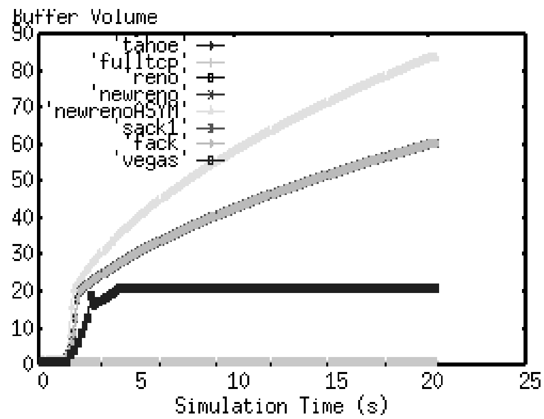

Fig. 4 Graph of buffer volume vs time

■ ***Algorithm to Calculate Buffer Size***

Input: {*time, Bcs, Tcs, Ts, cwnd_*}.

Bcs: current size of *buffer*.

Ts and *Tcs*: total and current simulation time.

*cwnd_*: variable of *Agent Class*.

Output: {*OFILE*: o/p file holding *time* and *buffer*}.

set *time*: =0.1

set *Bcs*: =current value of *cwnd_*

save in *OFILE* = [*time, Bcs*]

set *time*: = *time + Tcs*

iff *time*<*Ts* then goto step 2

Plot *OFILE* and EXIT.

Until the end of transmission, active buffer volume dynamically changes, shown by the curve family in Fig. 4 in which buffer utilization by NewrenoASYM is optimal.

E. Selecting the Best TCP

A sharp analysis of the figures above connotes the decision Table 4.

Table 4. The best TCP selection table

PARAMETERS	THE BEST TCP
Packet Vs Bandwidth	NewrenoASYM
Packet Vs Delay	Vegas
Packet Vs Rate	Sack1,Fack,NewrenoASYM
Buffer Vs Time	NewrenoASYM

III. CONCLUSION

Comparison of TCP variants with respect to other remaining network parameters would be an important future attempt. Such type of analysis is very helpful for selecting the appropriate TCP in a certain platform.

REFERENCES

[1] Nayama Islam, Sumyea Helal (2005), "Performance Analysis of TCP Tahoe, Reno, New Reno and SACK over Cellular Mobile System", 8^{th} *ICCIT*, Department of Information and Communication Technology, University of Rajshahi, Rajshahi-6205, Bangladesh, pp 786-790.

[2] NS, The network simulator-ns-2.27. *URL:http://www.isi.edu/nsnam/ns*

[3] Paul Meeneghan and Declan Delaney, "An Introduction to NS, Nam and OTcl scripting" National University of Ireland.

[4] Jae Chung and Mark Claypool, "NS by Example".

[5] Merida, Venezuela and ESSI, Sophia-Antipolis (2003), "NS Simulator for beginners", Lecture notes, 2003-2004, Univ. de Los Andes, France.

[6] "ns Tutorial", *http://www.isi.edu/nsnam/ns/tutorial/nsindex.html*

[7] "The ns Manual", The VINT Project, A Collaboration between researchers at UC Berkeley, LBL, USC/ISI, and Xerox PARC.

[8] Tanja Lang, (2002), "Evaluation of different TCP versions in non-wireline environments", The University of South Australia, Institute for Telecommunications Research.

[9] Johanna Antila, "TCP Performance Simulations Using Ns2", e-mail: *jmantti3@cc.hut.fi.*

[10] Hasegawa, Kenji Kurata and Masayuki Murata "Analysis and Improvement of Fairness between TCP Reno and Vegas for Deployment of TCP Vegas to the Internet", Graduate School of Engineering Science, Osaka University.

[11] Fall K. and Floyd, (1996), "Simulation based comparison of Tahoe, Reno, and Sack TCP ", *Computer Communication Review*, Vol. 26, No. 3, pp. 5-21.

[12] Lakshman, V, and Madhow, (1997), "The performance of TCP\IP for Networks with high Bandwidth Delay Products and Random loss", *IEEE Transaction On Networking*, Vol. 5, No.3.

[13] Michele Zorzi, A. Chockalingam, and Ramesh R. Rao, "Throughput Analysis of TCP on Channels with Memory", Senior Member, IEEE.

[14] HALA ELAARAG, "Improving TCP Performance over Mobile Networks", Stetson University.

[15] V. Jacobson, (1988),"Congestion Avoidance and Control", *Computer Communication Review*, vol. 18, no. 4, pp. 314--329.

[16] M. Miyoshi, M. Sugano and M. Murata, (2001), "Performance evaluation of TCP throughput on wireless cellular networks", *IEEE Vehicular Technology Conference (VTC)* , vol.3, , pp.2177 –2181.

[17] J. Mo, R.J. La, V. Anantharam, J. Walrand, (1999), "Analysis and comparison of TCP Reno and Vegas", *IEEE INFOCOM '99, Conference on Computer Communications*, vol.3, pp.1556-63.

ISBN: 978-988-17012-2-0