Comparative Performance Analysis of Versions of TCP in a Local Network with a Lossy Link

Anurag Kumar*

October 22, 1996

1 Introduction

In this work (see full report in [3]), we develop models and analyses for studying the bulk throughput of four versions of TCP; namely, OldTahoe (the original protocol), Tahoe, Reno, and NewReno [2]. We assume a local network scenario, in which a host on a local area network (LAN) is transmitting bulk data to a mobile host connected to the LAN by a wireless link. Our models incorporate important aspects, such as, *slow-start and congestion avoidance*, *coarse timers*, *fast-retransmit*, and *fast recovery*. Assuming an uncorrelated random packet loss model on the wireless link, we obtain the data throughput as a function of the packet error probability. Our results show how the throughput degrades, for each version, with increasing packet loss probability. For a given loss probability, our results quantify the performance improvement provided by each version.

As in [5] and [4], our analysis is also based on a study of the cyclical evolution of the TCP transmission process. Working with a Markov model, we analyse the congestion window process, as well as the duration of the cycle times. Formulating the problem as the reward rate in a Markov Renewal–Reward process [6], we can obtain the long run average bulk throughput of the protocol. In the local network scenario the major concern is not to keep the round trip pipe full, but to avoid a coarse timeout. The newer versions of TCP, namely, Tahoe, Reno and NewReno attempt to avoid coarse timeouts by trying to do a fast retransmit and fast recovery. Working with the distribution of the congestion window process at loss instants allows us to obtain the probability that fast recovery succeeds. Our model includes the duplicate acknowledgement threshold used by the fast-retransmit feature, and we can study the effect of reducing this threshold. A threshold of 1 is similar to one of the modifications to Reno proposed in the TCP-Vegas proposals (see [1]).

A simple packet loss model is assumed; with probability p a packet transmitted by the IS transmitter is lost; the losses are independent, i.e., we have a Bernoulli loss model. Typically, wireless links are subject to fading phenomena, which results in burst losses, rather than the uncorrelated losses that we have assumed here. This important extension is the subject of our current research.

*Current address: WINLAB, Rutgers University, Piscataway, N.J., 08854, U.S.A. (Permanent address: Dept. of Electrical Communication Engg., Indian Institute of Science, Bangalore, 560 012, INDIA.)

2 Numerical Results and Discussion

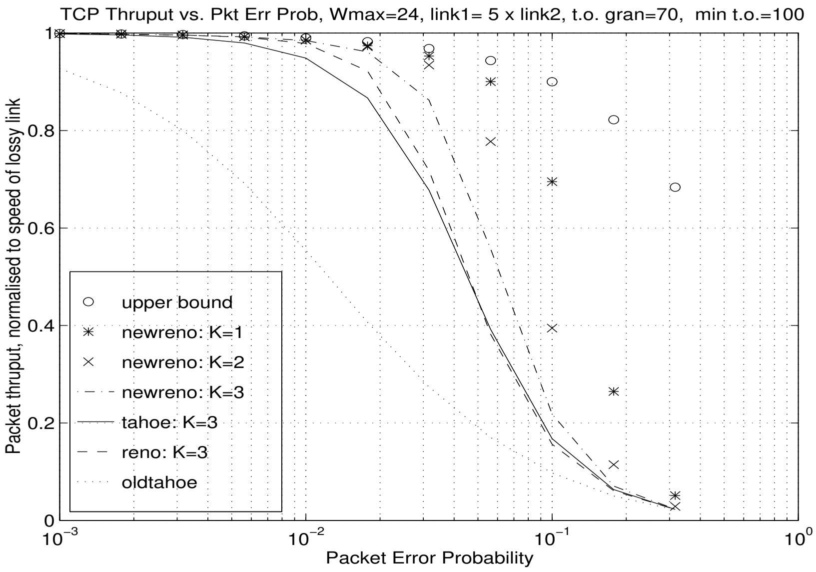

Figure 1: Throughput of versions of TCP vs. packet loss probability; (W_{max} = 24, LAN host transmission rate is 5 times the wireless link transmission rate, K is the fast-retransmit threshold).

Figure 1 shows a typical calculation from our exact analysis. All times are normalised to the mean packet transmission times at the lossy link. In these figures timeout granularity is 70, minimum timeout is 100, and the fast retransmit threshold $K \in \{1, 2, 3\}$; LAN host transmission rate is 5 times the rate of transmission on the wireless link, and the max window, W_{max}, = 24. Let us first put these numbers in perspective. We consider a wireless link of effective bit-rate 1.5Mbps; with 1400bytes TCP segments, this implies a packet transmission time of 7.5ms on the wireless link. Suppose that the LAN host, which is on a 10Mbps Ethernet, can source data at the rate of 7.5Mbps (5 times the wireless link rate); the reduction in rate being due to TCP processing. Assuming the BSD implementation of TCP, the timeout granularity is 500ms; i.e, about 70 times the mean packet service time. The minimum timeout is 2 "ticks", which implies an average of 750ms, or 100 times the mean packet transmission time on the lossy link. With 1400byte packets, $W_{max} = 24$ is equivalent to a receiver buffer sizes of about 32kB.

In this figure, throughput is plotted against packet loss probability. Since time is normalised to the service time on the lossy link, a throughput of 1 means that the lossy link is always occupied. We plot an upper bound that is an ideal throughput, not practically achievable by a distributed protocol.

The following is a summary of the main observations from our analysis and the numerical results (the actual numbers are for the parameters we have used):

1. With a minimum timeout of 100 times the mean service time at the lossy link, OldTahoe performs relatively poorly even with loss probabilities as low as .001, its throughput dropping to 0.5 of the lossy link service rate when the loss probability increases to .01.

2. The fast-retransmit feature in Tahoe, Reno and NewReno improves their performance significantly up to a loss probability of .01, but there is a precipitous drop beyond this point. Coarse timers, and a large minimum timeout, are the main reasons for this drop in throughput. Yet the throughput is more than twice that of OldTahoe, up to an error probability of about .15, beyond which all versions provide worse than 10% throughput.

3. NewReno derives the most advantage from the fast-retransmit feature, and increasing the maximum congestion window (W_{max}) helps NewReno the most. Since Reno is pessimistic in doing fast retransmission, and, in case of multiple losses, needs additional duplicate acks for continuing fast recovery, its throughput suffers for large packet loss probabilities; for $p > .05$ it is no better than Tahoe (with fast-retransmit). These results of ours corroborate the observations made by Fall and Floyd [2].

4. Setting the duplicate ack threshold for fast-retransmission to 1 can be expected to double the throughput of the basic fast-retransmit protocols for the error probability range $(.05, .2)$. This is related to one of the proposals in TCP-Vegas (see [1]).

5. Finally, we have observed that for the purpose of computing the throughput of the protocols with the fast-retransmit feature, it is important to capture the fast-retransmit probability well, and it is sufficient to use crude approximations for the mean cycle times. For the fast retransmission probability, we found it necessary to use the complete distribution of the congestion window at loss instants, whereas for the mean cycle times an approximation based on a simple fixed-point idea (also reported in [4]) worked well.

Subsequent to the work reported above, we have developed a throughput analysis of TCP with a correlated (Markov) packet loss model on the wireless link; the parameters of the Markov loss model are obtained from a Raleigh fading model.

References

[1] L. Brakmo and L. Peterson, "TCP Vegas: End-to-End Congestion Avoidance on a Global Internet," *IEEE Journal on Selected Areas in Communication*, Vol. 13, No. 8, pp. 1465-1480, Oct. 1995.

[2] K. Fall, and S. Floyd, "Comparisons of Tahoe, Reno, and Sack TCP," manuscript, ftp://ftp.ee.lbl.gov, March 1996.

[3] Anurag Kumar, "Comparative performance of versions of TCP in a local network with a lossy link," WINLAB Technical Report-TR129, Rutgers University, October 1996.

[4] T.V. Lakshman and U. Madhow, "Performance analysis of window-based flow control using TCP/IP: the effect of high bandwidth-delay products and random loss," *IFIP Transactions C-26, High Performance Networking V*, pp. 135-150, North-Holland, 1994.

[5] P.P. Mishra, D. Sanghi, and S.K. Tripathi, "TCP Flow Control in Lossy Networks: Analysis and Enhancements," in IFIP Transactions C-13 *Computer Networks, Architecture and Applications*, S.V. Raghavan, G.v. Bochmann, and G. Pujolle (eds.), Elsevier North Holland, pp. 181-193, 1993.

[6] R.W. Wolff, *Stochastic Modelling and the Theory of Queues*, Prentice Hall, 1990.