Effects of TCP congestion control mechanism on Self-Similarity of Network Traffic

Altyeb Altaher Altyeb
Department of Computer Science, International University of Africa
altypaltaher@yahoo.com

Abstract
A number of studies on traffic measurement from a variety of working networks have demonstrated that actual network traffic is self-similar in nature (traffic observed at different time resolutions has similar statistical properties and this phenomenon cannot be modeled well by traditional traffic models such as Poisson and Markovian processes). The causes of self-similarity of network traffic must be identified because understanding the nature of network traffic is critical in order to properly design and implement computer networks and network services. In this paper, the effects of TCP congestion control mechanism on traffic self-similarity are investigated.

I. INTRODUCTION

Recent studies have shown the presence of self-similarity in Ethernet LAN traffic [1], World Wide Web traffic [2], Wide Area Network traffic [3], etc. The issue of self-similarity has also been addressed in various studies from many different aspects including its effect on network performance [4], modeling techniques [4, 5], and causes of the appearance of self-similarity [6,7]. Since the pioneering work on self-similarity of network traffic by Leland et. al., many studies have attempted to determine the cause of this phenomenon. Initial efforts focused on application factors. For example, Crovella and Bestavros [8] investigated the cause of self-similarity by focusing on the variability in the size of the documents transferred and the inter-request time. They proposed that the heavy-tailed distribution of file size and "user thinks time" might potentially be the cause of self-similarity found in web traffic. Alternatively, few studies have considered the possibility that underlying network protocols such as TCP may cause or exacerbate the phenomenon [6,9, 10].

This paper is organized as follows. Section II describes the network model and network configuration. Results are presented and discussed In section III. Concluding remarks are given in section IV.

II. NETWORK MODEL

A. Network Model

Figure 1 illustrates the network model used in this work , the model is a server-client model, in which 32 clients are connected to two servers via a single link with a bandwidth of 10_Mbps.Network simulation package NS is used here ,with some modifications in codes. In the used model, each client requests a file transfer to a randomly selected server, and the server sends a file back as a series of fixed size (1K byte) packets. The following parameters are considered for the Ethernet: Max Propagation Delay 950_ns, Jam time after Collision 3.2 ms, Slot Size 51.2 ms , and Inter-frame Delay 9.6 ms

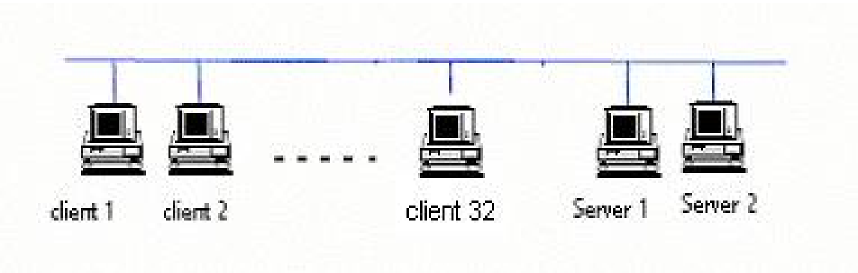

Figure 1: Network model

B. Signs of Self-Similarity

The above network has been configured to behave as sporadically congested network. The network alternates between congested and uncongested periods in order to examine the effects of TCP's congestion control mechanism on the self-similarity of network traffic.

Throughout this work , the Rescaled Adjusted Range Statistic (also called the R/S statistic) used to measure the degree of Traffic self-similarity.
Hurst Parameter (0.5 < H < 1.0) which measure the degree of self-similarity calculated as follow , for a Given set of observations X1,X2, X3, · · ·, Xn with

sample mean

$$\bar{x}(n) = \frac{1}{n}\sum_{i=1}^{n} x_i$$

sample variance

$$s^2(n) = \frac{1}{n}\sum_{i=1}^{n}\left(x_i - \bar{x}_n\right)^2$$

and

$$l_i(n) = \sum_{k=1}^{i} x_k - \bar{x}_n$$

the rescaled adjusted range (the R/S statistics) is defined as

$$\frac{R(n)}{S(n)} = \frac{\max_{1\le t\le n} l_i(n) - \min_{1\le t\le n} l_i(n)}{S(n)}$$

The Hurst parameter H is given by the equation

$$R(n)/S(n) \sim cn^H$$

Where the coefficient c was taken equal to 0.5 by Hurst

C. TCP congestion control mechanism

TCP is an adaptive mechanism that tries to utilize all free resources on the network. Adaptation is performed by the congestion control mechanism. Of course, full adaptation is not possible, as the network does not provide prompt and explicit information about the amount of free resources. TCP itself must test the path continuously by increasing its sending rate gradually until congestion is detected, which is signaled by a packet loss, and then it adjusts its internal state variables accordingly.

III. The RESULTS

In order to see the effect of TCP congestion control algorithm on the self- similarity of network traffic, the total traffic load offered was varied by varying the number of clients and the Hurst parameter of the traffic generated by the clients was calculated, the network congestion status was divided into three levels :

1. Uncongested: the amount of traffic generated is much lower than the available bandwidth.
2. Moderately congested: the amount of traffic generated causes some, but not severe, congestion.
3. Heavily congested: the amount of traffic generated is higher than what the network can handle.

A quantitative measure of self-similarity is obtained by using the Hurst parameter H which expresses the speed of decay of a time series autocorrelation function. Hurst parameter which measure the degree of self-similarity calculated by the relation :

$$E[R(n)/S(n)] \approx cn^{H}, n \rightarrow \infty$$

Table 1 shows H as a function network congestion status .It was notice that Hurst parameter vary with varying the congestion status.
In the uncongested case, hurst parameter takes the smallest value which indicates low degree of network traffic self similarity. This result is due to the absence of congestion in the network, i.e., the congestion control mechanism has not activated to control or modulate the network traffic. When the clients generate a moderate amount of traffic, and hence introduce intermittent congestion, the TCP congestion control mechanism begins to modulate the network traffic, because the TCP changes its congestion window periodically when triggered by a packet loss event.

Traffic Self similarity	0.70	0.73	0.75
Congestion Status	Uncongested	Moderately congested	Heavily congested

Table 1. Hurst parameter estimates based on (R/S) at three different network status.

In order to investigate the effect of TCP congestion control algorithm on the self-similarity of network traffic ,the number of transmitted packets are observed at different time resolutions and it's statistical properties are compared. For this purpose , the fluctuation of the congestion window size was traced by defining the interval from the (i − 1)-th packet loss event to the i-th packet loss event as the i-th cycle. further the i-th cycle was divided into RTTs and consider a congestion window size in each RTT. The congestion window size of TCP at the j-th RTT of the i-th cycle was defined as W(i, j).

The congestion control mechanism of TCP consists of two phases: the slow start phase and the congestion avoidance phase. in each of which TCP uses a different algorithm in increasing/decreasing congestion window size. In the slow start phase, TCP increases its window size by one packet on receiving an ACK packet. On the other hand, in the congestion avoidance phase, TCP increases its window size W(i, j) by 1/W (i, j) packets when it receives an ACK packet. Focusing on the changes in congestion window in a RTT to investigate the effect of the rapid fluctuation of the congestion window size on the network traffic self-similarity , the congestion window size W (i, j) is derived as follows:

$$W(i,j) = \begin{cases} (Slow\_start\_Phase:) \\ 2 * W(i,j-1) \xrightarrow{if} W(i,j-1) < S(i) \\ (Congestion\_Avoidance\_Phase:) \\ W(i,j-1)+1 \xrightarrow{if} W(i,j-1) \geq S(i) \end{cases}$$

where S (i) is a slow start thresh (ssthresh) value in the i-th cycle at which TCP changes its phase from the slow start phase to the congestion avoidance phase.

We can see the effect of tcp congestion control algorithm in network traffic self-similarity in Table 1 as the number of client increase the hurst parameter increased to 0.73, under heavy congestion, the hurst parameter increased to 0.75 .and hence indicate that the congestion control mechanisms of TCP noticeably modulate traffic when the network is moderately congested; that is, TCP induces self-similarity into the network traffic.

Figure 2 shows time series plots of network traffic measured in packet per time unit, as a function of time. The figure shows plots which span three orders of magnitude in time scale and three different network congestion status. The time units used vary from 20 second in the left most column to 60 sec in the right most column. The three rows show how traffic varies when the network congestion status vary from uncongested in upper row to Heavily congested in lower row. It worth to notice that higher network congestion generates greater traffic self similarity.

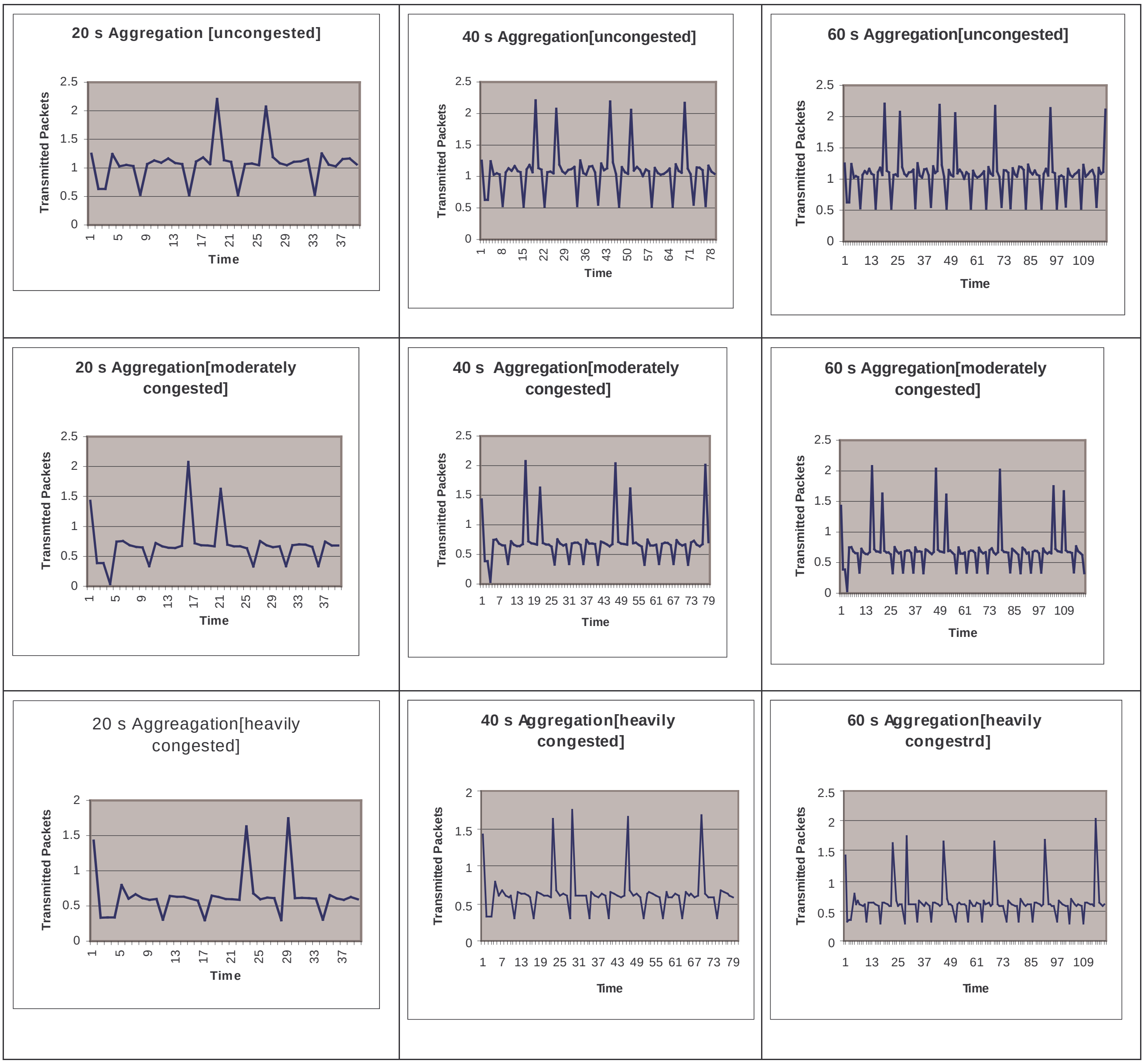

Figure 2 : plots which span three orders of magnitude in time scale and three different network congestion status.

IV Conclusions

In this paper ,the effect of tcp congestion control algorithm on network Traffic Self similarity is considered. The result indicates that the congestion control mechanism of TCP modulates the network traffic to be self similar, this modulation occurs for two primary reasons: (1) the rapid fluctuation of the congestion window size caused by the continual “additive increase / multiplicative decrease (or re-start slow start)” probing of the network state and (2) the dependency between the congestion control decisions made by TCP congestion control mechanism.

References

1. W.E. Leland, M.S. Taqqu, W. Willinger and D.V. Wilson, “On the self similarity nature of Ethernet traffic,” in Proc. ACM SIGCOMM, pp. 183-193, Sept. 1993.
2. M.E. Crovella and A. Bestavros, “Self-similarity in World Wide Web traffic:Evidence and possible cause,” IEEE/ACM Trans. on Networking, vol.6, pp.835-846, Dec. 1977.
3. V. Paxson and S. Floyd, “Wide-Area Traffic: The failure of Poisson modeling. IEEE/ACM Trans. on Networking, vol. 3, no. 3, pp. 226-44, June 1995.
4. K. Park, G. Kim, and M. Crovella, “On the effect of self-similarity on network performance,” in Proc. of the SPIE International Conf. on Performance and Control of Network System, pp 296-310, Nov 1997.
5. A. Ost and R.H. Boudewijn, “Modeling and Evaluation of pseudo selfsimilar traffic with Infinite-State Stochastic Petri Nets,” in Proc. of the workshop on formal method and telecom., pp. 120-136, Sept 1999.
6. J. M. Peha, “Protocols can make traffic appear self-similar,” in Proc. of the1997 IEEE/ACM/SCS Comm. Networks and Distributed System. Modeling and Simulation Conf., pp. 47-52, Jan 1997.
7. A. Veres and M. Boda, “The chaotic nature of TCP congestion control,” inProc. IEEE INFOCOM 2000, Tel Aviv, Israel, pp. 1715-1723, Apr 2000.
8. M.E. Crovella and A. Bestavros, “Self-similarity in World Wide Web traffic: Evidence and possible cause,” IEEE/ACM Trans. on Networking, vol.6, pp. 835-846, Dec. 1977.
9. D. R. Figueiredo, et. al., “On the autocorrelation structure of TCP traffic,” Comp. Networks Journal, Special Issue on Advances in Modeling and Eng. of Long-Range Dependent Traffic, vol. 40, no. 3, pp. 339-361, Oct. 2002.
10. L. Guo, M. Crovella, I. Malta, “How does TCP generate pseudo-selfsimilarity?” in Proc. of MASCOTS '01, Cincinnati, Ohio, Aug. 2001.

Article received: 2005-12-24