The Effect of TCP on the Self-Similarity of Network Traffic [1]

Biplab Sikdar and Kenneth S. Vastola
Department of ECSE
Rensselaer Polytechnic Institute
Troy, NY 12180 USA
email: {bsikdar,vastola}@networks.ecse.rpi.edu

Abstract — **Research on the causes of self-similarity in network traffic has till recently focused primarily on the application level and human factors. However, protocol specific causes, specially the fact that TCP can lead to long-range dependence has become apparent in the recent past. In this paper we show how TCP's retransmission and congestion control mechanism, specifically its timeout and exponential backoff mechanism, can lead to self-similarity in aggregate TCP flows. We develop a mathematical formulation which shows that TCP's underlying algorithms result in packet dynamics of a TCP flow being analogous to a number of ON/OFF sources with OFF periods taken from a heavy tailed distribution. Using well known limit theorems, we then show that this leads to the self-similar nature of TCP traffic. Our mathematical model shows a direct correlation of the loss rates to the degree of self-similarity. Measurements on traces collected by us also exhibit this relationship predicted by our model. Our results also show that the loss rate can be used a representation of the effect of the network and the superposition of multiple flows.**

I. Introduction

Research on the causes of self-similarity in network traffic have primarily focused on the application level dynamics of high-speed networks and the human factors involved while the effect of the protocol dynamics and the network have received attention only in the recent past. In [17], the causes of the self-similarity are investigated at the source level. In [1] the authors cite the distribution of file sizes, the effects of caching and human factors like response time and preference as possible causes for the self-similarity in WWW traffic. It was pointed out in [9] and [2] that closed loop protocols like TCP lead to much richer scaling behavior than open loop protocols like UDP.

In this paper, we investigate the effect of TCP on the self-similarity of network traffic. We also account for the effects of the network in terms of the losses it introduces and the multiplexing of flows in a path. In [15], the authors attribute the self-similarity of TCP traffic to the chaotic nature of TCP's congestion control mechanism. The adaptive nature of TCP's congestion control is suggested as the cause for the propagation of self-similarity in the Internet in [16]. The main aim of our paper is to understand the effects of TCP's retransmission and congestion control mechanism on the observed self-similarity of TCP traffic. We show that the traffic generated by a *single* TCP connection can exhibit self-similarity in contrast to all previous work (except for [15]) which concentrated on the aggregate traffic. In addition, we show that the degree of self-similarity has a direct relationship with the losses experienced by a flow with the traffic no longer self-similar, i.e. $H \approx 0.5$ for very low loss rates. While similar phenomena have been reported recently (after this paper was completed), their models to explain the self-similarity either require unrealistic loss rates to induce self-similarity [5] or are able to show long-range dependence over very small time scales [4]. In this paper, we present a model of TCP based on ON/OFF processes which explains the self-similarity of TCP traffic and validate it using TCP traces collected from the Internet. We also give a mathematical formulation of how TCP's congestion control mechanism leads to self-similarity in the traffic it generates and account for the effects of the network in terms of the loss probabilities and the presence of other flows.

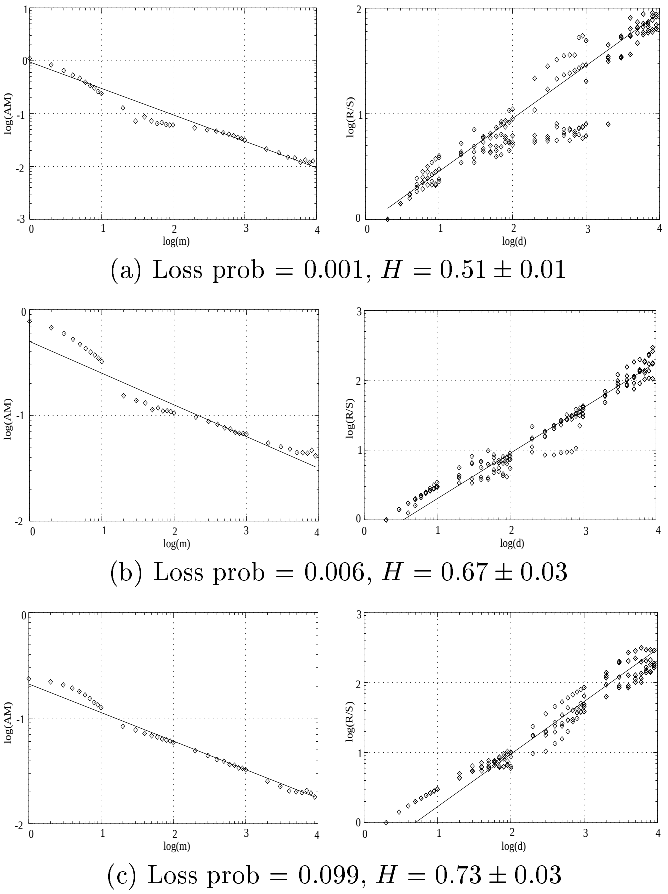
This paper reports the main results from a larger work available as a technical report [12] which contains details on the derivations and more results. The rest of the paper is organized as follows. In Section II we first present results of tests for the presence of self-similarity in individual TCP transfers over the Internet. We then present and validate a model which explains this self-similarity. Section III provides a mathematical foundation for our model and investigates the mechanisms of TCP which lead to the self-similarity. Finally, Section IV presents the discussions and concluding remarks.

II. Self-similarity of TCP Flows

In this section we provide experimental evidence of the self-similarity of individual TCP flows which motivates the investigation of TCP dynamics for causes of self-similarity. In [15] *ns* simulations were used to show that the data sent by a single TCP flow in the superposition of a number of TCP flows shows evidence of self-similarity in contrast to all previous studies which concentrated on the aggregated traffic. To verify the self-similar nature of single TCP microflows in actual Internet transfers, we first present the results from tests for long-range dependence on traces collected from real life TCP connections over the Internet.

The traces were collected for TCP transfers originating from a machine running Solaris 2.6 at Troy, NY. The destinations for the transfers were in Columbus, OH (HP-UX), Los Angeles, CA (FreeBSD Cairn-2.5), Boston, MA (Linux 2.0.36) and Pisa, Italy (FreeBSD 3.3). Due to space restrictions, we show results for only the transfers Italy. The results for the others are similar and are presented in [12]. Each trace is 2000 seconds or around 33 minutes long and was collected using `tcpdump` which did not lose any packets. The transfers were done over periods in 1999 and 2000 at various times of the day and week. Depending on the prevalent network conditions, the loss rates experienced by each flow is different and

[1] Supported in part by DARPA contract F30602-00-2-0537 and in part by DoD MURI contract F49620-97-1-0382.

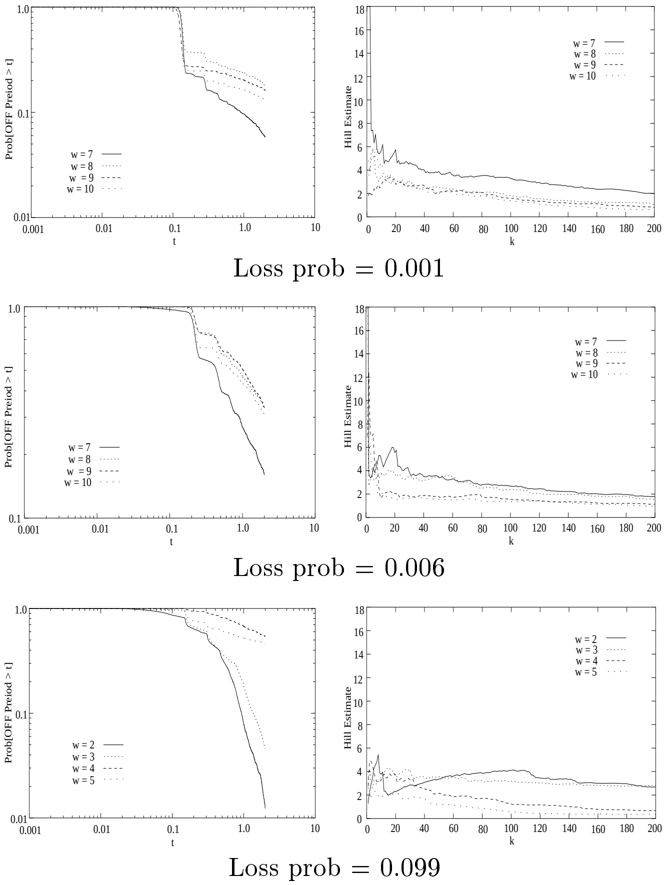


(a) Loss prob = 0.001, $H = 0.51 \pm 0.01$

(b) Loss prob = 0.006, $H = 0.67 \pm 0.03$

(c) Loss prob = 0.099, $H = 0.73 \pm 0.03$

Figure 1: Tests for self-similarity: Absolute value method (left) and the R/S statistics method (right).



Loss prob = 0.001

Loss prob = 0.006

Loss prob = 0.099

Figure 2: Tests for heavy-tailed nature of the OFF times: ccdf plots (left) and Hill's estimates (right) for various values of w.

we use this to classify transfers between a source-destination pair.

Figure 1 shows the results of the tests for long-range dependence on three traces to Pisa, Italy which had loss rates of 0.001, 0.006 and 0.099. We tested for long-range dependence using three of the widely used methods: the absolute value method, R/S statistics method and the periodogram method and show results for only the first two methods. The results clearly show the long-range dependence in the individual TCP flows. Also, the degree of long-range dependence, as indicated by the Hurst parameter, is clearly dependent on the loss rate experienced by the flow, with higher loss rates leading to larger values of H. Also note that for extremely low probabilities (less than 0.001) the traffic is no longer self-similar with $H \approx 0.5$ as shown in section (a) of Fig. 1.

This poses the following questions. What are the underlying mechanisms responsible for the direct influence of the loss probabilities of the self-similarity of TCP traffic? Can the effects of the network and the influence of the superposition with other flows be abstracted using the single parameter of the loss probability? And most importantly, what is TCP's role in all this? In this paper we address these issues and show how TCP's retransmission and congestion avoidance mechanisms contribute to the self-similar nature of network traffic.

A ON/OFF Model Based Explanation and its Validation

To give an explanation for TCP's effect on the self-similarity of network traffic, we consider a TCP flow to be composed of the superposition of W_{max} ON/OFF processes. Each process corresponds to each of the possible values that the *cwnd* of the flow might have since W_{max} is the receiver's advertised maximum buffer size and is the upper limit on *cwnd*. A *cwnd* of w, corresponding to the w^{th} ON/OFF process, $1 \leq w \leq W_{max}$, implies a deterministic ON time which is equal to the time to transmit the w packets with the packets generated at a constant rate during this period. We note that though in practice there might be a small variation in the time between two successive packets in a round, with high speed networks and ACK compression these variations are negligible when compared to the RTTs.

The OFF period for the w^{th} process, $1 \leq w \leq W_{max}$, corresponds to the time interval between two successive instants where *cwnd* has the value w. Now, if the distribution of these times has a heavy tail, their complementary cumulative distribution function (ccdf) $F_c(x)$ behaves like

$$F_c(x) \sim l x^{-\alpha} L(x) \qquad \text{with } 1 < \alpha < 2 \tag{1}$$

where $l > 0$ is a constant, $L(x)$ is a slowly varying function at infinity, i.e., $\lim_{x \to \infty} L(tx)/L(x) = 1, \forall\, t > 0$ and the relation $f(x) \sim g(x)$ implies $\lim_{x\to\infty} f(x)/g(x) = 1$. We can now use Theorem 2 of [14] which says that the superposition of a number of these processes converges in the limit to fractional Brownian motion (fBm) and thus exhibit self-similarity.

In our case, the limiting conditions are reached when we have a large number of flows in the network each contributing its ON/OFF processes to the superposition. Now we just need to show that the distribution of the OFF times indeed corresponds to the form of Eqn. 1. In Fig. 2 we plot the ccdf of the OFF times for various window sizes for the traces for Italy and the heavy tailed nature of each is clearly evident. A statistically more rigorous method for estimating the slope of the tails and thus α as compared to the eyeballing method associated with plotting ccdfs is the *Hill's estimator* [6]. The presence of heavy tails is indicated by a straight line behavior of the Hill's estimate $\hat{\alpha}_n$ as the number of samples used in the calculation of the estimate increases while a steadily decreasing pattern is a strong indication of the data being not from a heavy-tailed distribution. Fig. 2 also plots the Hill's estimates for the OFF time distribution for various window sizes for the Italy traces and clearly they are consistent with the form of Eqn. 1. Thus we can conclude that the superposition of such ON/OFF process from a number of TCP flows will converge in the limit to fBm and thus exhibit self-similarity.

It is interesting to note the ccdf and Hill estimate plots for the Italy trace with $p = 0.001$. From Figure 1 $H \approx 0.5$ for this trace, i.e. the trace does not exhibit self-similarity. We note from Figure 2 that the Hill estimates for all the ON/OFF process corresponding to this trace are decaying constantly and thus do not have a heavy tailed nature thereby failing to satisfy the conditions of Theorem 2 of [14]. As a result the trace is not self-similar. In Section C from our derivation of a lower bound of the ccdf it will be clear why low loss rates fail to give rise to heavy tails.

An important assumption here is the independence of the window sizes of different flows, which need not be the case for *all* the flows in a link. Simulation studies have indicated that the window sizes of TCP flows sharing a common bottleneck link may get synchronized though such synchronization is hard to observe in the Internet [7]. Also, most of the simulation studies focus on very heavily congested bottleneck links while link loads in practice tend to be comparatively much lower. Also, note that the independence requirements fail to be satisfied only when nearly all the flows in a link are correlated. To prove that the independence assumptions of Theorem 2 of [14] are satisfied, we analyzed some of the traces reported in [10]. The results of our statistical tests on these traces to see if the individual TCP flows are indeed independent indicate that amongst the longer flows in the traces, roughly 35-70 % of the flows are mutually independent, providing enough independent flows in the superposition.

An important part in the calculation of the OFF times is what criterion we use to define a OFF period. We define an ON period to be over whenever the distance between two successive packets in the trace exceeds a length δ dependent on the packet transmission time on the link. By keeping δ sufficiently small we can ensure that the spacing between the packets in the ON period is almost constant thus satisfying the requirement of Theorem 2 of [14]. Also, as in [17], the exact numerical choice of δ does not affect the results and the heavy tailed nature of the ccdf remains an invariant independent of the choice of δ.

III. Investigating the Role of TCP

We now pinpoint the sources in TCP's retransmission and congestion avoidance mechanism which are responsible for the self-similarity of network traffic. We then derive a lower bound on the tail of the OFF time distribution and show that it decays according to a power law providing a firm mathematical foundation to our model. In this paper we concentrate on TCP Reno as it the most widely deployed variant of TCP. The effect of the other versions of TCP is discussed in Section IV. We assume that the reader is familiar with the basic concepts of TCP like the congestion window *cwnd*, slow start, delayed acknowledgments etc and refer the reader to [13] for details on TCP's algorithms.

A The Impact of Timeouts

From the explanation for the observed self-similarity in TCP traffic given in Section II it is obvious that the central aspect of the phenomenon lies in the infinite variance or the heavy tailed nature of the OFF time distributions. Let us now consider the features of TCP which lead to such a behavior.

In the following we assume an infinite or steady state flow currently in the congestion avoidance mode to make the visualization easier. The occurrence of heavy tails in the OFF times is mainly due to the losses which lead to timeouts. This is due to the following reasons. A timeout represents a significant duration when no packets are transmitted and acts as a boundary between ON and OFF periods of the flow as a whole leading to a bursty nature of TCP traffic. The durations of timeouts are generally an order of magnitude greater than the RTT [8] and with coarse TCP timer granularities and variations in the RTT measurements can be quite large. Again, if the retransmitted packet following a timeout is also lost, the silent period is doubled and from the traces reported in [8] the occurrence of multiple consecutive timeouts is frequent. Also, a majority of the losses experienced by TCP flows lead to timeouts which can be attributed to the fact loss that most routers in the Internet deploy droptail queues. Correlated loss models, where all the packets following the first dropped packet in a round are also dropped are an appropriate models for the losses arising from these queues [8]. This coupled with the fact that a single loss in a window less than 4, two or more losses in a window less than 8 and three or more losses for higher windows in TCP Reno will lead to a timeout contributes to the large proportion of timeouts in the observed loss indications. Before moving on to the derivation of the lower bound on the tail of the ccdf, we first derive the probability that a loss in a window of size w leads to a timeout.

B Probability of Timeouts

Consider a round with window w and let the probability that a loss of any packet in this round will lead to a timeout be denoted by $Q(w)$. We assume that the receiver uses delayed ACKs. We also assume droptail queues and the correlated loss model of the previous subsection. Packet losses in a round are assumed to be independent of losses in other rounds and the packet loss probability is denoted by p.

For window sizes less than 4, any packet loss leads to a timeout and thus $Q(w) = 1$ for $1 \leq w \leq 3$. For windows with $4 \leq w \leq 8$ two or more packet losses in a round leads to a timeout. If only one packet is lost in the current round, if we lose any packet in the following round, the flow will eventually timeout. In addition the retransmitted packet must also be transmitted successfully to avoid a timeout. Thus $Q(w)$ for this range of window values is given by

$$Q(w) = 1 - \frac{p(1-p)^{2w-1}}{1-(1-p)^w} \qquad \text{for } 4 \leq w \leq 8 \tag{2}$$

For window sizes greater than 8, three or more losses in a round will lead to a timeout. Also we have to ensure that the retransmitted packet is received successfully along with the fact that none of the packets in the succeeding round are lost. Neglecting the extremely few possibilities in which it is possible to recover a single loss in the succeeding round without going into a timeout, we have

$$Q(w) = 1 - \frac{p(2-p)(1-p)^{2w-2}}{1-(1-p)^w} \qquad \text{for } 9 \le w \le W_{max}$$

C A Lower Bound on the OFF Time Distribution

We now derive a lower bound on the ccdf by identifying the possible ways in which the time between two successive windows of the same size can exceed a given value. In this derivation, we measure time in units of the round trip time.

Let us assume that the current window size is w and we want to find the probability that the time until the next instant where *cndw* $= w$ is greater than 100. The most obvious possibility is that the flow does not experience any loss for the next 100 rounds so that after some round the *cwnd* stays at W_{max}. Another possibility could be that after i rounds (when *cwnd* $> 2w$) the flow experiences a loss which results in a fast retransmit. The flow then transmits the next $100 - i$ rounds without any loss. As a variation of this we could have a number of successive fast retransmits without reaching a window of w. Yet another line of possibilities is timeouts. Let us denote the average duration of a timeout (in terms of RTTs) by $E[TO]$. As the first possibility we could have that there are no losses in the first $100 - E[TO]$ followed by a timeout. We could also have i initial rounds without loss and then n timeouts (with n sufficiently large) before the window gets a chance to increase to w. Other possibilities include cases where we have timeout periods of length $2E[TO]$, $4E[TO]$ and so on. Each of these cases represent independent possibilities whose individual contribution to the tail of the OFF time distribution has an exponential decay, the rate of which depends on the corresponding probability of the loss indications and their effects.

The tail of the OFF time distribution for each window size and the corresponding ON/OFF process can thus be seen as the superposition of a large number independent exponential tails each with its own rate of decay. The mix of these independent exponentials leads to a composite distribution which has a heavy tail over the region of our interest. The following theorem by Bernstein [3] can be used to show that the superposition of a number of properly chosen exponentials can be used to model heavy tailed distributions like Pareto and Weibull in the region of primary interest.

Theorem 1. (Bernstein) *Every completely monotone pdf f is a mixture of exponential pdfs, i.e., $f(t) = \int_0^\infty \lambda e^{-\lambda t} dG(\lambda)$, $t \ge 0$ for some proper cdf G.*

We now obtain the probabilities corresponding to each of the possible paths that we described.

Case 1: The no loss case. Consider the w^{th} ON/OFF process which corresponds to a *cwnd* of w, $1 < w < W_{max}$. Assume that the current round has a window of size w. The probability that the next window of size w occurs after t RTTs, assuming there are no losses in between, is given by

$$P\{T > t\} = (1-p)^{N(t)} \tag{3}$$

where $N(i)$ represents that number of packets that are transmitted in the i rounds following the round with size w and is given by

$$N(i) = \begin{cases} iw + \lceil \frac{i}{2} \rceil \left(i - \lceil \frac{i}{2} \rceil\right) & \text{if } i \le j \\ jw + \lceil \frac{j}{2} \rceil \left(j - \lceil \frac{j}{2} \rceil\right) + (i-j)W_{max} & \text{else} \end{cases} \tag{4}$$

where $j = 2(W_{max} - w) - 1$.

Case 2: Fast retransmission losses. Consider again the w $^{\text{th}}$ ON/OFF process, $1 < w < W_{max}$. We can have a OFF time greater than t if we have loss indications at windows greater than $2w$ which result in fast retransmits. For simplicity, we consider only those cases where the loss occurs in a window of size W_{max}. The flow first transmits packets without loss for the first i rounds during which its window reaches W_{max}. It then experiences a loss which is recovered by a fast retransmit. Since $w < \lceil W_{max}/2 \rceil$ the desired window size is not achieved at the beginning of the congestion avoidance mode. Also, following each loss there are $2(W_{max} - m) - 1$ rounds with $W_{max}(W_{max} - 1) - m(m+1)$ packets till *cwnd* reaches W_{max} again with $m = \lceil W_{max}/2 \rceil$. Thus there are $t - n - n(2(W_{max} - m) - 1) - 2(W_{max} - w) + 1$ rounds with successfully transmitted windows of W_{max}. The total number of correctly transmitted packets, after algebraic simplifications, is thus

$$\begin{aligned} N_c(w,t) = {} & W_{max}(t - (n+1)W_{max} + 2w + 2nm - 4n) \\ & -w(w+1) - nm(m-1) \end{aligned} \tag{5}$$

Now, since there are $M = t - 2nW_{max} + 2w + 2(n-1)m - 2n + 3$ rounds with a *cwnd* of W_{max} with n of them having losses, the probability that the OFF time is greater than t is given by

$$\begin{aligned} P\{T > t\} \; = \; & \begin{pmatrix} M \\ n \end{pmatrix} (1 - (1-p)^{W_{max}})^n \\ & (1 - Q(W_{max}))^n (1-p)^{N_c(w,t)} \end{aligned} \tag{6}$$

Also, since each loss is associated with $2(W_{max} - m) - 1$ rounds where the window is not W_{max}, the maximum possible losses in t rounds can be shown to be limited by

$$n_{max} = \left\lfloor \frac{t - 2(W_{max} - w) + 1}{2W_{max} - 2m - 1} \right\rfloor \tag{7}$$

Case 3: Loss indication resulting in a timeout. Consider the case when the loss occurs after i rounds from the round with a window of w. The number of packets transmitted in these i rounds, $N(i)$ is given in Eqn. 4 and the value of the *cwnd* in the i^{th} round w_i is given by

$$w_i = \min\{W_{max}, w + \lceil i/2 \rceil\} \tag{8}$$

The number of packets transmitted in the slow start phase which follows a timeout, $t_{ss}(w_i)$ is obtained using the model of [11] which is more accurate than the commonly used approximation where the window always increases 1.5 times every RTT.

$$t_{ss}(w_i) = \left\lfloor 2\log_2\left((2m)/(1+\sqrt{2})\right)\right\rfloor - 1 \tag{9}$$

where $m = \lceil \frac{w_i}{2} \rceil$ and the number of packets transmitted in the slow start phase can be expressed as

$$N_{ss}(w_i) = \left\lfloor 2^{\frac{t_{ss}(w_i)+1}{2}} + 3 \cdot 2^{\frac{4t_{ss}(w_i)-3}{8}} - 2 - \frac{3\sqrt{2}}{2} \right\rfloor \tag{10}$$

If $w > m$ we also have a linear phase where the window increases linearly from m to w. The total time required by the flow to reach a window of w again following the timeout is thus

$$D_{nl}(w, w_i) = \begin{cases} t_{ss}(w) + E[TO] + 1 & \text{if } w \leq m \\ t_{ss}(w_i) + E[TO] + 2(w-m) & \text{else} \end{cases} \tag{11}$$

Now, the probability that we have a loss in a round of size u following the timeout, before the window reaches w, $P_{TO}(u, w_i)$, $1 \leq u < w$, is given by

$$P_{TO}(u, w_i) = \begin{cases} (1-p)^{N_{ss}(2u)}(1-(1-p)^u)Q(u) & \text{if } u < m \\ (1-p)^{N_{ss}(w_i)}(1-(1-p)^{2u})Q(u) & \text{else} \\ (1-p)^{u(u-1)-m(m-1)} & \end{cases} \tag{12}$$

Then, the probability that there is another timeout before the window reaches a window of w is given by

$$P_s(w, w_i) = \sum_{u=1}^{w-1} P_{TO}(u, w_i) \tag{13}$$

After the i^{th} round, on an average 2 more rounds of packets are sent (where two losses are recovered) before the timeout period begins. Thus if $i \geq t - D_{nl}(w, w_i) - E[TO] - 2$, the probability that the OFF time is greater than t is given by

$$P\{T > t\} = \begin{cases} (1-p)^{N(i)}(1-(p)^{w_i})Q(w_i) & \text{if } i \geq I_l \\ (1-p)^{N(i)}(1-(p)^{w_i}) & \text{else} \\ Q(w_i)(1-P_s(I_l - i)) & \end{cases}$$

where $I_l = t - E[TO] - 2$.

Case 4: Loss of the retransmitted packet. When the retransmitted packet following a timeout is also lost, the retransmission timer backs off exponentially with a factor of 2 leading to very large silent periods. The duration of a sequence of n consecutive losses in lengths of $E[TO]$ is given by

$$L_n = \begin{cases} 2^n - 1 & \text{for } n \leq 6 \\ 63 + 64(n-6) & \text{else} \end{cases} \tag{14}$$

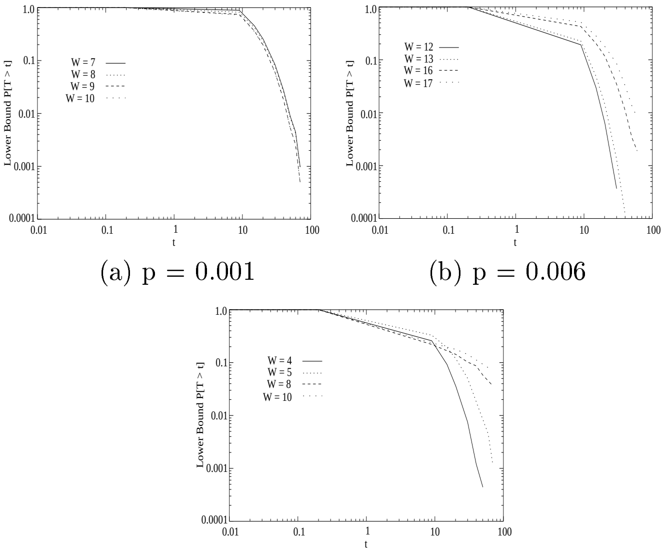
Each of the losses following the initial loss indication occur with probability p. Also, the linear phase of the *cwnd* following the second loss begins after *cwnd* reaches 2. Then, if $i > t - L_n E[TO] - 2(w-2) - 1$ the probability that the off time for window w is greater than t is given by

$$P\{T > t\} = \begin{cases} (1-p)^{N(i)}(1-(1-p)^{w_i})Q(w_i)p^{n-1} & \text{if } i \geq I_l \\ (1-p)^{N(i)}(1-(1-p)^{w_i})Q(w_i)p^{n-1} & \text{else} \\ (1-P_s(I_l - i)) & \end{cases}$$

where $I_l = t - L_n E[TO] - 2$.

Case 5: n isolated timeouts. Let us now consider the case where there are n isolated timeouts each of length $E[TO]$. After the first loss after i rounds, the slow start phase lasts till *cwnd* reaches $m = \lceil \frac{w_i}{2} \rceil$. All subsequent losses occur before *cwnd* reaches a values of w. The expected duration between the first and the second loss indications is given by

$$D_l(w_i) = \begin{cases} E[TO] + 2 + \frac{1}{1-P_s(w-1)} & \text{if } w < m \\ \left(\sum_{u=2}^{w-1} uP_{TO}(u)\right) & \\ E[TO] + 2 + \frac{1}{1-P_s(w-1)} & \text{else} \\ \left(\sum_{u=2}^{m-1} uP_{TO}(u) + \sum_{u=m+1}^{w-1}\right. & \\ \left.(u + 2(u-m) - 0.5)P_{TO}(u)\right) & \end{cases}$$



(a) p = 0.001

(b) p = 0.006

(c) p = 0.099

Figure 3: Lower bound on the ccdf for the Italy traces for various values of w. The time t is in seconds.

Similarly, we model the average duration between two successive losses by $D_l(w)$. After the last loss, it takes $t_{ss}(w-1) + 2(w - \lceil \frac{w-1}{2} \rceil) - 1$ rounds for the window to reach a size of w. Since $t - D_l(w_i) - i$ rounds comprise the duration for the rest of the losses following the first loss indication, we need at least $n = \lceil (t - D_l(w_i) - i)/(D_l(w)) \rceil + 1$ losses for the off time to exceed t. Then if $n > 1$ (the case $n = 1$ had already been considered) the probability that the off time is greater than t is given by

$$P\{T > t\} = \begin{cases} (1-p)^{N(i)}(1-(1-p)^{w_i}) & \text{if } i \geq I_l \\ Q(w_i)P_s(w, w_i)(P_s(w, w))^{n-2} & \\ (1-p)^{N(i)}(1-(1-p)^{w_i}) & \text{else} \\ Q(w_i)P_s(w, w_i)(P_s(w, w))^{n-2} & \\ (1-P_s(I_l - i)) & \end{cases}$$

where $I_l = t - D_l(w_i) - (n-2)D_l(w) - E[TO] - 2$.

Case 6: Multiple consecutive losses. We now consider the cases where there are n losses which are successfully recovered using a single timeout and l losses which lead to exponential backoffs. Let the l periods of consecutive timeouts be all due to j consecutive losses. The probability of each of these l periods is $P_s(w, w)p^{j-1}$ and the probability of the single loss indications is $P_s(w, w_i)$ and $P_s(w, w)$ for the first and the rest of the $n-1$ losses respectively. For a given n and l we can have a sequence corresponding of $n + l$ losses in t rounds only if $t - D_l(wi) - (n + l - 1)D_l(w) - l(2^j - 2)E[TO] \leq i < t - D_l(wi) - (n + l - 2)D_l(w) - (l-1)(2^j - 2)E[TO]$. For the values of i falling in this range, the probability that the off time is greater than t is given by

$$P\{T > t\} = \begin{cases} (1-p)^{N(i)}(1-(1-p)^{w_i})Q(w_i) & \text{if } i \geq I_l \\ P_s(w, w_i)(P_s(w, w))^{n+l-2}p^{l(j-1)} & \\ (1-p)^{N(i)}(1-(1-p)^{w_i})Q(w_i) & \text{else} \\ P_s(w, w_i)(P_s(w, w))^{n+l-2}p^{l(j-1)} & \\ (1-P_s(I_l - i)) & \end{cases}$$

where $I_l = t - D_l(w_i) - (n + l - 2)D_l(w) - l(2^j - 2)E[TO] - E[TO] - 2$.

D Numerical Results

In Fig. 3 we show the numerical evaluation for the lower bounds on the ccdf for the parameters from all the Italy traces

Type of Loss	$p = 0.100$		$p = 0.001$	
	prob	ccdf	prob	ccdf
Case 1	0.0000	0.0000	0.0304	0.0304
Case 2	0.0000	0.0000	0.0000	0.0304
Case 3	0.0000	0.0000	0.0123	0.0428
Case 4	3.99E-4	3.99E-4	3.18E-6	0.0428
Case 5	0.0116	0.0120	0.0156	0.0584
Case 6	0.1306	0.1426	3.57E-5	0.0584

Table 1: The contribution of various losses to the ccdf. t is 200 RTTs, $w = 10$ and $W_{max} = 18$.

considered in Section II. The heavy tailed nature of the tails is evident and as expected, the rate of decay reduces with increasing loss probabilities. Also, to see the impact of timeouts on the tails of the ccdf, in Table 1 we show the contribution to the tails by the various cases involving timeouts that we considered in the previous subsection. As expected, the contribution from the timeouts have a large contribution to the tails, specially higher loss probabilities. For very low loss rates, the contribution due to multiple losses is negligible and the tail is made of just 3-4 exponentials. For higher losses, the probability of multiple timeouts increases and we have a large number of exponentials with different rates the superposition of which leads to a heavy tailed distribution.

IV. Conclusions and Discussions

In this paper we provided an explanation of how TCP can cause self-similarity in network traffic. Using traces of actual TCP transfers over the Internet, we showed that individual TCP flows, isolated from the aggregate flow on the link also have a self-similar nature. Our results also showed that the degree of self-similarity is directly proportional to the loss rates experienced by the flow. We then proposed a model explaining this self-similarity and presented empirical evidence supporting it showing that each TCP flow can be considered as the superposition of a number of ON/OFF processes. We also provided a firm mathematical basis to the empirical observations of heavy-tailed distributions in the OFF times by deriving a lower bound on the ccdf.

A natural construction of the extremely bursty nature of TCP traffic comes from timeouts which represent "silent" periods and separate periods of activity. Since a majority of loss indications under current Internet scenarios lead to timeouts, losses increase the burstiness and the heavy tails in the OFF times. The degree of self-similarity or H being dominated by the heaviest tail in the superposition, higher loss rates thus lead to higher values of H. In contrast, when the loss rate is extremely low TCP transmits W_{max} packets in every round and behaves like a CBR source and the traffic is longer self-similar. This explains the observations in Section II where flows with loss rates less than 0.001 had a Hurst parameter of approximately 0.5. Our findings and the calculations for the lower bound on the OFF time distribution show that the loss probability is a faithful indicator of the "network's effect" on TCP traffic in terms of both the effects of superposition with other flows and the degree of self-similarity of the traffic.

While TCP Reno is the most widely implemented version of TCP, other versions are currently under research, the most notable amongst them being TCP SACK. TCP SACK provides robustness against multiple packet losses in a single window and recovers them without resorting to timeouts. However, it does not completely eliminate timeouts since it requires the receipt of K (usually 3) duplicate ACKs before the retransmission mechanism kicks in. Thus timeouts are inevitable for small windows and will be present even for larger windows for correlated losses. Consequently we expect self-similarity to be present in TCP SACK traces also, though the loss rates at which $H > 0.5$ will be greater than those for TCP Reno.

References

[1] M. Crovella and A. Bestavros, "Self-similarity in World Wide Web traffic: Evidence and possible causes," *IEEE/ACM Transactions on Networking,* vol. 5, no. 6, pp. 835-846, Dec 1997.

[2] A. Feldmann, A. C. Gilbert and W. Willinger, "Data networks as cascades: Investigating the multifractal nature of Internet WAN traffic," *Computer Communications Review,* vol. 28, no. 4, pp. 42-58, 1998.

[3] W. E. Feller, *An introduction to probability theory and its application,* Wiley, New York, 1971.

[4] D. R. Figueiredo, B. Liu, V. Misra and D. Towsley, "On the autocorrelation structure of TCP traffic," Technical Report TR 00-55, University of Massachusetts, Computer Science Department, Amherst, MA, 2000.

[5] L. Guo, M. Crovella and I. Matta, "TCP congestion control and heavy tails," Technical Report BU-CS-2000-017, Boston University, Computer Science Department, Boston, MA, July 2000.

[6] B. M. Hill, "A simple general approach to inference about the tail of a distribution," *Annals of Statistics,* vol. 3, pp. 1163-1174, 1975.

[7] M. May, T. Bonald and J.-C. Bolot, "Analytic evaluation of RED performance," *Proceedings of IEEE INFOCOM,* pp. 1415-1424, Tel-Aviv, Israel, March 2000.

[8] J. Padhye, V. Firoiu, D. Towsley and J. Kurose, "Modeling TCP Reno performance: A simple model and its empirical validation," *IEEE/ACM Transactions on Networking,* vol. 8, no. 2, pp. 133-145, April 2000.

[9] K. Park, G. Kim, and M. Crovella, "On the relationship between file sizes, transport protocols, and self-similar network traffic," *Proceedings of International Conference on Network Protocols,* pp. 171-180, Columbus, OH, October 1996.

[10] V. Paxson and S. Floyd, "Wide area traffic: The failure of Poisson modeling," *IEEE/ACM Transactions on Networking,* vol. 3, no. 3, pp. 226-244, June 1995.

[11] B. Sikdar, S. Kalyanaraman and K. S. Vastola, "Analytic models for the latency and steady-state throughput of TCP Tahoe, Reno and SACK," Preprint, 2001.

[12] B. Sikdar and K. S. Vastola, "Issues in TCP and self-similarity of network traffic," Technical report ECSE-NET-2001-2, Networks Laboratory, Department of ECSE, Rensselaer Polytechnic Institute, Troy, NY 2001.

[13] W. R. Stevens, *TCP/IP illustrated volume 1,* Addison Wesley, 1994.

[14] M. S. Taqqu, W. Willinger and R. Sherman, "Proof of a fundamental result in self-similar traffic modeling," *Computer Communication Review,* vol. 27, no. 2, pp. 5-23, April 1997.

[15] A. Veres and M. Boda, "The chaotic nature of TCP congestion control," *Proceedings of IEEE INFOCOM,* pp. 1715-1723, Tel-Aviv, Israel, 2000.

[16] A. Veres, Z. Kenesi, S. Molnar and G. Vattay, "On the Propagation of Long-Range Dependence in the Internet," *Proceedings of ACM SIGCOMM,* pp. 243-254, Stockholm, Sweden, September 2000.

[17] W. Willinger, M. S. Taqqu, R. Sherman and D. V. Wilson, "Self-similarity through high-variability: Statistical analysis of Ethernet LAN traffic at the source level," *IEEE/ACM Transactions on Networking,* vol. 5, no. 1, pp. 71-86, February 1997.