# The origin of TCP traffic burstiness in short time scales

Hao Jiang
Georgia Tech
hjiang@cc.gatech.edu

Constantinos Dovrolis
Georgia Tech
dovrolis@cc.gatech.edu

*Abstract*— **Internet traffic exhibits multifaceted burstiness and correlation structure over a wide span of time scales. Previous work analyzed this structure in terms of heavy-tailed session characteristics, as well as TCP timeouts and congestion avoidance, in relatively long time scales. We focus on shorter scales, typically less than 100-1000 milliseconds. Our objective is to identify the actual mechanisms that are mostly responsible for creating bursty traffic in those scales. We show that TCP self-clocking, joint with queueing in the network, can shape the packet interarrivals of a TCP connection in a two-level ON-OFF pattern. This structure creates strong correlations and burstiness in time scales that extend up to the Round-Trip Time (RTT) of the connection, especially for bulk transfers that have a large bandwidth-delay product relative to their window size. Also, the aggregation of many flows, without rescaling their packet interarrivals, does not converge to a Poisson stream, as one might expect from classical superposition results. Instead, the burstiness in those scales can be significantly reduced by TCP pacing, depending however on the minimum pacing timer. Finally, we show that sub-RTT burstiness is important in queueing performance not only in moderate load conditions, as previously shown, but also in high loads when the bottleneck buffer size is relatively small.**

## I. Introduction

To characterize and explain the variability of Internet traffic has been a long research endeavor. The discovery of Self-Similarity (SS) and Long-Range Dependency (LRD) in both LAN and WAN traffic in [1] was a breakthrough, as it established that Internet traffic exhibits a strong correlation structure that extends up to several hours (LRD signature), and that the traffic process, scaled by an appropriate factor, maintains the same distribution when averaged across a wide range of time scales (SS signature) [2]. The significance of these statistical properties for queueing performance and traffic prediction has been debated, with some models arguing for a major impact (see [3]), while others insisting that finite buffers limit the largest correlation scale[1] that can affect queueing (see [4]). Another major step was the identification of the physical mechanisms that cause LRD and SS behavior in large scales: it is the heavy-tailed distribution of the duration of active or idle user times [5], also related to the distribution of transfer sizes and user thinking times [6].

More recently, another question raised significant attention: what is the impact of TCP, the dominant transport protocol, on the correlation structure of Internet traffic? Contrary to earlier claims that TCP can create self-similarity, it has been now established that the correlations introduced by TCP with its retransmission timeout and the congestion avoidance algorithm extend over a certain range of time scales, from a few Round-Trip Times (RTTs) up to tens or hundreds of seconds [7], [8], Nevertheless, these scales are of great interest in traffic prediction and capacity provisioning, and so the effect of these TCP mechanisms should not be ignored in traffic modeling or simulation studies.

The major open question in the quest to understand Internet traffic, however, is the burstiness of Internet traffic in “short” scales. Short, here, typically refers to time intervals up to 100-1000 milliseconds, in which the factors that cause large-scale correlations may not yet be strong or even present. Previous work in this area has given somewhat contradictory results both in characterizing the variability of Internet traffic through statistical models, but also in identifying the mechanisms that create that variability. A review of previous results is given later in this section. For now, we note that the traffic models which have been proposed cover the range from a simple Poisson process, independent Gamma interarrivals, to quite complex non-Gaussian multifractal processes. We emphasize that even though the previous models are quite different, they do not have to be wrong. It is possible that different models capture the variability of Internet traffic with different degrees of accuracy, or that different network links carry traffic with widely different characteristics.

Our objective in this paper is to examine the link between the TCP protocol and the short scale burstiness of Internet traffic. The emphasis on TCP is justified by the fact that typically more than 90% of Internet traffic is carried by the TCP protocol. Additionally, since we are interested in short time scales, we focus on the TCP self-clocking mechanism. Self-clocking is responsible for shaping the interarrivals of data segments from a single flow in *sub-RTT scales*, i.e., in time intervals that are shorter than the RTT of that flow. Specifically, we start with the following basic question: can a TCP flow create bursty traffic in sub-RTT scales, and if so, under which conditions?

We find that TCP self-clocking, joint with network queueing due to the packets of the flow itself or due to cross traffic, can shape the packet interarrivals of a TCP flow in a one-level or two-level ON-OFF pattern, respectively. This packet-train

This work was supported by the “Scientific Discovery through Advanced Computing” program of the US Department of Energy (award number: DE-FG02-02ER25517), by the “Strategic Technologies for the Internet” program of the US National Science Foundation (award number: 0230841), and by an equipment donation from Intel. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the previous funding sources.

[1] We use the terms “time scale” and “scale” interchangeably.

structure creates strong correlations and burstiness in the sub-RTT scales, especially in the case of bulk transfers that have a large bandwidth-delay product relative to their window size. Such flows can build up a large window, due to their size. Additionally, if their bandwidth-delay product is quite larger than their window, they tend to transmit their packets in bursts, or clusters of bursts.

We also show that the aggregation of many flows, without rescaling their packet interarrivals, does not converge to a Poisson stream, as one might expect from classical superposition results. Instead, the correlation structure of individual flows can shape the correlation structure of the aggregate stream, independent of how many flows are multiplexed together. Looking for ways to reduce the burstiness of Internet traffic in short scales, we show that ideal TCP pacing at the sources is very effective. Instead of only examining ideal pacing, however, we also consider the presence of a minimum pacing timer and show that that timer would have to be in the order of 1ms for pacing to be effective in practice.

Finally, we show that sub-RTT burstiness is important in queueing performance. Previous work showed, using infinitely long buffers, that short time scale burstiness is only important in moderate utilizations, while the presence of LRD dominates queueing performance in heavy utilizations. Extending that work, we show that short time scale variability is also important in heavy loads as long as the bottleneck buffer size is relatively short.

We should note that the objective of this work is not to propose a new traffic model for the short time scale burstiness of Internet traffic. Also, we do not claim that TCP self-clocking is the only mechanism that can create such burstiness. Our results can help, however, to explain the results of previous measurement studies in terms of the characteristics of the TCP flows that were dominant in the corresponding packet traces.

**Previous work in short scale effects***:* [9] was one of the first papers to focus on short time scales, and also to use wavelet-based multiresolution analysis to characterize the *scaling behavior* (a special type of correlation structure) of Internet traffic. [10] provided empirical evidence that WAN traffic can be modeled using a *multifractal model*, similar to that developed in [11]. More recent work, however, argues that the traffic at a tier-1 ISP is well modeled as monofractal, rather than multifractal [12]. This discrepancy is probably due to differences in the marginal distribution of the traffic used by the two studies: if the marginal distribution is Gaussian, which is often the case with traces from WAN links, the process can only exhibit monofractal scaling [13].

[14] showed that scaling in short time scales is related to the TCP closed-loop flow control, and argued that the cutoff between short and long scale behavior is, roughly, the RTT of the TCP transfers. [14] identified *ACK compression* [15], [16], which is a specific case of TCP self-clocking failure, as a primary suspect for the scaling behavior observed in short time scales. Our work provides extends the results of [14], explaining why the RTT is a natural boundary between short and long scale correlations, and showing the conditions under which self-clocking produces bursty traffic in sub-RTT scales.

[17] established that the short scale burstiness does not depend on the TCP flow arrival process. Additionally, in not heavily loaded networks, correlations across different flows do not affect the short scale burstiness either. In a follow-up work, [18] showed that the the correlation structure of aggregate traffic can be captured by a Poisson cluster process in which the packet interarrivals within individual clusters of each flow follow an overdispersed Gamma distribution, while the flow volumes are heavy-tailed.

[12] introduced the concept of “dense flows” i.e., flows with bursts of densely clustered packets, and showed that dense flows create short time scale burstiness. Our work explains the presence of dense flows based on TCP self-clocking. [19] showed that short time scale burstiness can have a significant impact on queueing performance, especially in moderate utilizations, while correlations in coarser scales are more important in heavy utilizations. It should be emphasized however that [19] considers an infinite-buffer queueing system, which tends to overemphasize the importance of correlations in large scales (LRD).

[20] proposed an interesting classification of traffic in *alpha* and *beta* flows. The former are large transfers over high-capacity links and produce bursty traffic in short time scales, while the latter are are mostly low-throughput or short transfers and they produce Gaussian and LRD traffic. [21] identified nine ways in which a TCP or UDP source can send long back-to-back packets (referred to as “source-level bursts”), causing significant correlations in short scales. Those reasons include UDP message segmentation, TCP slow start, lost ACKs, and others. A queueing model that models TCP source-level bursts with a Markovian batch arrival process was recently proposed in [22].

[23] showed that Web flows, which are relatively short compared to bulk transfers, produce a linear relation between the mean and the variance of the traffic process in the 100ms time scale. Such a linear relation is characteristic of a Poisson process, but only for the given scale. In an extension of the previous work, [24] showed that the variance-mean relation depends on the network load and on the time scales of interest, and that a general characterization of the traffic as “Poisson-like” would not be accurate.

More recently, [25] argues that network traffic today can be well represented by the Poisson model in sub-second scales. The authors explain their findings based on a certain interpretation of a classical result from the theory of point processes, namely that, as the number of multiplexed flows increases, the aggregate traffic process tends to a Poisson process [26]. A similar claim has been made in [27], but for the case that the traffic load increases as the number of multiplexed flows increases.

**Paper overview***:* Section II gives a brief background on wavelet-based multiresolution analysis, and it defines what we mean by *burstiness* at a particular time scale more precisely. Section III shows that TCP self-clocking and network queueing can create a one-level or two-level ON-OFF structure in the packet interarrivals of a TCP flow, under a certain condition, in sub-RTT scales. Section IV explains that flow aggregation cannot make the traffic uncorrelated, if the constituent flows have a correlation structure. Section V serves as a case-study, in which we analyze an OC-48 trace and identify the flows that shape the sub-RTT burstiness of the aggregate stream. Section VI examines the effect of pacing, as a practical way to make TCP traffic smoother at the sources. Finally, Section VII evaluates the impact of short scale burstiness in the delay and loss rate performance of a finite-buffer queue. We conclude in Section VIII.

## II. Energy plots and burstiness

In this section, we describe a statistical tool that we use throughout the paper to analyze the burstiness of a traffic process in a range of time scales. This tool is based on *wavelet-based MultiResolution Analysis (MRA)*, and it was developed by Abry and Veitch [28], [29]. The traffic process at a network link can be described as a time series of packet arrival times and sizes. More commonly, a traffic process is described as a sequence of counts that measure the amount of bytes appearing at the link in successive and non-overlapping intervals of a certain duration. Specifically, the *counting process* at a time scale $T_j$=$2^j T_0$ ($j = 0, 1, \ldots$) is a time series $X_j$=$\{X_{j,0}, X_{j,1}, \ldots\}$, with $X_{j,k}$ the amount of bytes in the $k$'th interval $t_{j,k}$ of duration $T_j$. The scale $T_0$ is our *reference time scale*, and it corresponds to the minimum interval in which counts are measured.

Informally, the term “burstiness” refers to the statistical variability of the traffic process at a given scale $T_j$. High variability in $X_j$ implies a more fluctuating traffic load, when the latter is measured at scale $T_j$. Since there is no particular scale that we should be only interested in, the variability of $X_j$ is typically measured and analyzed in a wide range of time scales. Even though a number of statistical techniques can be used for measuring the variability of a traffic process, such as the Index of Dispersion for Counts, Index of Dispersion for Intervals, or the Power Spectral Density, we prefer to use wavelet-based MRA energy plots, produced with [29], following a number of previous studies in this area that adopted the same technique.

An MRA energy plot shows the variance of the wavelet coefficients of the traffic process $X_j$ as a function of the scale index $j$. An important assumption is that $X_j$ is *covariance stationary*, meaning that, for a given $j$, the mean of $X_j$ is constant and the covariance between any $X_{j,k}$ and $X_{j,k'}$ only depends on $|k - k'|$. In the following, we limit the presentation in the special case of *Haar wavelets*. The Haar wavelet coefficients $W_{j,k}$ at a scale $j$ are defined as

$$W_{j,k} = 2^{-j/2}(X_{j-1,2k} - X_{j-1,2k+1}) \qquad (1)$$

The *energy* $\mathcal{E}_j$ at scale $j$ is then defined as the variance of the coefficients $W_{j,k}$,

$$\mathcal{E}_j = \mathrm{Var}[W_{j,k}] = 2^{-j} E[(X_{j-1,2k} - X_{j-1,2k+1})^2] \qquad (2)$$

or,

$$\mathcal{E}_j = 2^{-j}\mathrm{Var}[\Delta X_{j-1,k}] \qquad (3)$$

where $\Delta X_{j-1,k}$=$X_{j-1,2k}$-$X_{j-1,2k+1}$ (with $E[\Delta X_{j-1,k}]$=0). Equation (3) gives an interpretation of the energy $\mathcal{E}_j$ at scale $j$: it is the variance of the traffic variation $\Delta X_{j-1,k}$ at scale $j-1$, scaled by the factor $2^{-j}$. A more common interpretation, related to the power spectral density of the time series $X_{j,k}$, can be found in [28].

In practice, the energy $\mathcal{E}_j$ is estimated from a finite time series as

$$\mathcal{E}_j \approx 2^{-j} \frac{\sum_{k=1}^{N_j} (\Delta X_{j-1,k})^2}{N_j} \qquad (4)$$

where $N_j$ is the number of wavelet coefficients at scale $j$. An *energy plot*, like that of Figure 1, shows the base-2 logarithm of $\mathcal{E}_j$ as a function of $j$. Note that the top of the graph shows the time scale $T_{j-1}$ (in milliseconds), while the corresponding scale at the x-axis is $j$. The reason for this mismatch is that $\mathcal{E}_j$ is determined by the terms $\Delta X_{j-1,k}$ at the previous scale $T_{j-1}$.

The MRA signature of a *Poisson process* (independent exponential interarrivals) is that its energy plot is a horizontal line. This can be easily proven as follows. Due to the memoryless property of the exponential distribution, the process $X_j$ is independent at any scale $j$, and so

$$\mathrm{Var}[\Delta X_{j-1,k}] = 2\,\mathrm{Var}[X_{j-1}] = 2^j \mathrm{Var}[X_0]$$

where $\mathrm{Var}[X_0]$=$\lambda T_0$ is the variance of a Poisson process with rate $\lambda$ at a time scale $T_0$. So, the energy of $X_j$ at any scale $j$ is

$$\mathcal{E}_j = 2^{-j}\, 2^j \mathrm{Var}[X_0] = \lambda T_0$$

meaning that *the energy plot of a Poisson process is a horizontal line at* $\log_2(\lambda T_0)$.

The Poisson process plays a major role in this paper, providing a reference point for the burstiness of other traffic processes. Since, traditionally, the Poisson process has been considered as a benign traffic model in terms of queueing performance, we say that *a traffic process $X_j$ is bursty at scale $j$* if the energy of $X_j$ is higher than the energy of a Poisson process with the same average rate. Otherwise, we say that *$X_j$ is smooth at scale $j$*.

A reader that is familiar with previous MRA studies will notice that we use energy plots in a different manner than [28]. Specifically, most previous works focused on the *scaling behavior* of the traffic process, which is characterized by the slope of $\mathcal{E}_j$ in a range of time scales. We focus, instead, *on the burstiness of the traffic process relative to a Poisson process of the same average rate, in different time scales*. Consequently, we are interested in the absolute magnitude of $\mathcal{E}_j$, rather than in its local slope. The reason we focus on burstiness relative to Poisson traffic, rather than on scaling behavior, is that the former is clearly linked to the well-known statistical and queueing characteristics of Poisson processes. Furthermore, a traffic process may exhibit local scaling behavior, or more generally it may have a strong correlation structure in a range of scales, but without being bursty (i.e., without being burstier than the Poisson process). For example, a periodic process is strongly correlated, but at the same time it is the smoothest among all traffic models.

In the following, we show some energy plots for various synthetic traffic processes. Out objective is, first, to provide insight in the interpretation of MRA plots, and second, to show the energy plots for some models that we use later in the paper. Figure 1 shows the energy plots for three

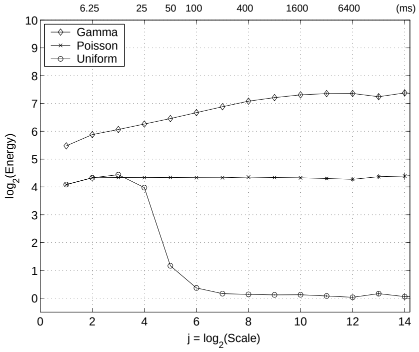


Fig. 1. Energy plots for Exponential, Gamma, and Uniform interarrivals.

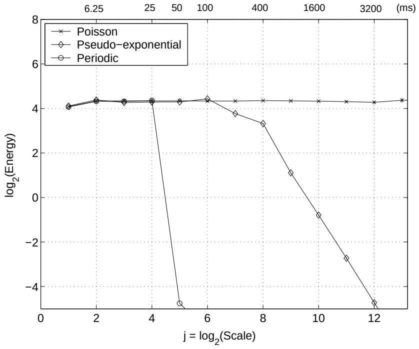


Fig. 2. Energy plots for Periodic and Pseudo-exponential interarrivals.

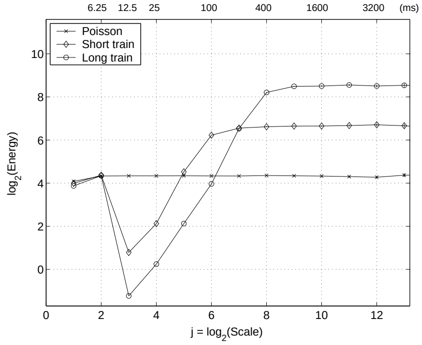


Fig. 3. Energy plots for two packet train models.

renewal processes (i.e., independent and identically distributed interarrivals). The corresponding interarrival distributions are Exponential, Gamma, and Uniform. All three distributions have the same average interarrival (50ms), while the reference scale is $T_0$=25ms. Because the average rate is the same in all three processes, they would asymptotically have the same energy at scale $T_0$ if the latter would tend to zero (in infinitesimal time scales all point processes with the same rate look the same).

Note that even if the interarrivals are independent, the counting process $X_j$ can be correlated [30]. Then, the variance Var$[\Delta X_{j-1,k}]$ can increase faster than $2^j$ due to positive correlations between $X_{j-1,2k}$ and $X_{j-1,2k+1}$. For example, Figure 1 shows the energy plot of independent Gamma interarrivals with shape parameter $c$=1/4. Notice that this process is bursty, in the sense that it has a higher energy than the Poisson process in all scales.

On the other hand, Figure 1 also shows the case of Uniform interarrivals in a range [L,U], with $L$=30ms and $U$=70ms. The lower and upper bounds on the interarrivals limit the minimum and maximum number of arrivals, and thus the variability of $X_j$, in any scale $T_j$>$L$. This explains why the energy of $X_j$ is lower than the energy of the corresponding Poisson process, making the Uniform process smooth in time scales larger than $L$. It is also easy to see that the energy difference between the Uniform and Poisson processes increases as the range $U$-$L$ decreases.

Reducing the range of the uniform distribution to $[T-\epsilon, T+\epsilon]$, where $\epsilon$ is very small compared to $T$, leads to a practically periodic process with period $T$. The energy of a periodic process becomes zero theoretically (and its logarithm drops to $-\infty$) at the scale that corresponds to $T$, because $\Delta X_{j-1,k}$=0 for all $k$ at that scale. This is shown in Figure 2 with $T$=50ms. The periodic model can be viewed as the smoothest traffic process, as it has the minimum possible energy after scale $T$.

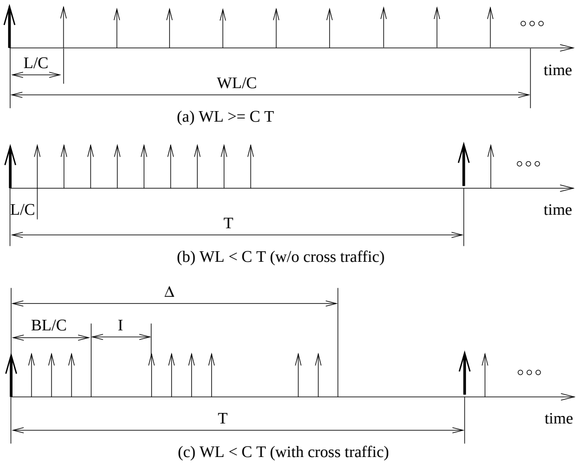
Later in the paper, we consider a periodic source that sends $W$ packets in each time interval of length $T$, with the $W$ packets randomly distributed within that interval. One way to model this source is to generate $W$ independent exponential random numbers $(r_1, r_2, \ldots r_W)$ with mean $T/W$, and then scale them up or down by a certain factor $c$ so that their sum is equal to $T$. We refer to this process as *Pseudo-Exponential* with parameters $(W, T)$. Figure 2 shows the energy plot of a Pseudo-Exponential process with $W$=8 and $T$=400ms. As expected, the energy drops dramatically after the time scale $T$, due to the corresponding periodicity. In lower scales, the traffic is almost as bursty as the Poisson process, but with a slightly lower energy due to the variability reduction imposed by the constraint $\sum_{i=1}^{W} cr_i = T$.

Another traffic process that we use extensively in the rest of the paper is the *packet train* model [31]. In that model, an *ON-period* consists of $W$ packets that arrive with a constant interarrival $\tau_o$, causing a burst of duration $W\tau_o$. After every burst, an *OFF-period* follows, which is exponentially distributed with mean $\tau_f$. The average rate of the process is $W/(W\tau_o + \tau_f)$ packets per second. Figure 3 shows the energy plot for two packet train models with $(W, \tau_o, \tau_f)$ set to (8, 100ms, 300ms) and (32, 400ms, 1200ms), respectively. The following facts can be easily for the packet train model directly through (2). First, the packet train model has a periodicity at $\tau_o$, which becomes stronger as $W$ increases, causing an energy drop at the corresponding scale. Then, the energy increases from $\tau_o$ to $W\tau_o$ with a slope of 2.0. The energy in time scales larger than $W\tau_o$ remains constant, and higher than the energy of the corresponding Poisson process, meaning that the packet train model generates bursty traffic in those scales. Also, for a given average rate, the energy in time scales larger than $W\tau_o$ increases with the train length $W$ due to the presence of longer bursts.

## III. A sub-RTT model of TCP self-clocking

The operation of a TCP sender in sub-RTT scales, namely the transmission of a given window worth of data during each round-trip, is largely determined by the self-clocking mechanism. According to self-clocking, TCP should typically send a new packet whenever it receives a new ACK [32]. In the case of Delayed-ACKs, which is the norm today, an ACK is generated for every second received packet, and so the sender usually generates a packet pair for every received ACK[2]. An important implication of self-clocking is that the sender does not need to schedule packet departures based on a timer; packets are transmitted on an event basis, as ACKs arrive from the receiver. The timing of ACKs, however, is determined by a number of effects, including queueing of the data packets in the forward path, random delays at the receiver, as well as queueing of the ACKs in the reverse path. Consequently, the sender has no direct control on the timing of packet departures, and so, under certain conditions, it can send a large number of packets at a rate that is much higher than the flow's average throughput.

In this section, we present a simple model of self-clocking that explains how a TCP sender can be trapped into a state of sending long packet trains at the full capacity of the forward path. The model will be presented incrementally in two parts: first, ignoring any cross traffic in the forward path, and second considering such traffic. We make several simplifying assumptions: ACKs are not affected by queueing in the reverse path, there is only a single queueing point in the forward path (single bottleneck), and the sender and receiver do not introduce delays in the transmission of data packets and ACKs, respectively. Even with these assumptions, however, we show with MRA energy plots that the burstiness of the traffic generated by our model is quite similar to the burstiness of real TCP flows in sub-RTT scales. Consequently, the effects that we ignore with the previous assumptions may be of secondary importance compared to the effect of forward path queueing on self-clocking.



(a) WL >= C T

(b) WL < C T (w/o cross traffic)

(c) WL < C T (with cross traffic)

Fig. 4. Packet interarrivals for a TCP flow during a particular round-trip. Three cases are shown depending on whether $WL<CT$, and on whether there is cross traffic in the flow's bottleneck.

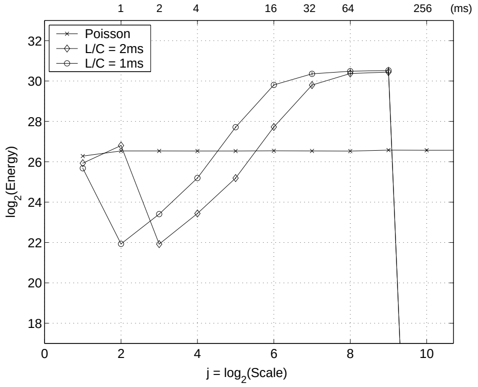


Fig. 5. Energy plot for TCP self-clocking model without cross traffic.

### A. Self-clocking without cross traffic

Consider a TCP flow with a minimum RTT $T$ seconds. The capacity of the bottleneck link $\mathcal{B}$ at the forward path is $C$ bytes/sec. The *bandwidth-delay product* of the flow is defined as $CT$ bytes. The flow's RTT is larger than $T$ when a queue builds up at $\mathcal{B}$. Suppose that all data packets have a size of $L$ bytes. We initially ignore the effect of Delayed-ACKs; it is easy to modify the model later so that it considers Delayed-ACKs.

Since there is no cross traffic, an initial window of $W_0$ packets sent back-to-back will arrive at the receiver periodically, with a *dispersion* (i.e., time-spacing) $\tau$=$L/C$ between any two successive packets. The receiver will respond to each packet with an ACK, sent with the same dispersion $\tau$. $T$ seconds after the sender had sent its first packet, it will receive the first ACK and it will start sending the packets of the second round-trip, with a larger window size $W_1$. This process will repeat in the following round-trips, until the flow reaches the maximum window allowed by the socket buffers, or until it experiences a congestion event or timeout. The key point, however, is that during each round-trip the TCP sender sends its current window $W_k$ with a dispersion $\tau$ between successive packets, i.e., at the full capacity $C$ of the forward path. With Delayed-ACKs, an ACK is generated in every $2\tau$ seconds, but the dispersion between successive data packets of the same round-trip is still $\tau$ because an ACK releases two back-to-back packets in that case.

Suppose now that, during a particular round-trip, the flow has a send-window of $W$ packets. Based on the previous timing analysis, we know that the sender transmits the $W$ packets with a period $\tau$ at the start of the round-trip. If $W\tau\geq T$, the flow saturates the path achieving the maximum possible throughput $C$, and the effective RTT increases to $WL/C$. In that case, the traffic process is periodic with period $L/C$, as shown in Figure 4-a, and the energy plot becomes as in Figure 2. So, when the window of a TCP flow (in bytes) is larger than the flow's bandwidth-delay product, i.e., when $WL\geq CT$, the traffic that TCP generates is extremely smooth.

On the other hand, it is easy to see that if $W\tau<T$, the traffic process follows the packet train model of §II. Specifically, TCP generates a packet train in every round-trip with a length of $W$ packets, dispersion $\tau_o$=$\tau$=$L/C$ during the train, and with an idle time $\tau_f$=$T$-$W\tau$ between trains. The time-pattern of this traffic process is shown in Figure 4-b. Notice that TCP does *not* distribute its window throughout the RTT. Instead, when

[2] There are several deviations from this behavior, including congestion window increases, idle times when the application has no data to send, and retransmission timeouts.

there is no cross traffic, the entire window appears back-to-back at the full capacity $C$ of the forward path in the start of the corresponding round-trip, creating a one-level ON/OFF packet structure.

Figure 5 shows the energy plot of the previous model for $T$=256ms, $W$=16 packets, $L$=1250 bytes, and for two values of $\tau$: 1ms ($C$=10Mbps), and 2ms ($C$=5Mbps). Note that $W\tau$ is 16ms or 32ms, which is less than $T$ in both cases. The energy plot of a Poisson process with the same average rate $WL/T$ is also shown for comparison. The major observation in these energy plots is that *if the TCP window (in bytes) is less than the flow's bandwidth-delay product, TCP generates bursty traffic in the sub-RTT scales that extend between approximately $WL/C$ and $T$.*

Also note that for given $T$, $W$, and $L$, and thus for a given average rate $WL/T$, the range of scales in which the traffic is bursty increases with $C$. This implies that *if the capacity of a path increases, but with a constant TCP average throughput, the source creates more bursty traffic in sub-RTT scales*. Finally, the energy drop at scale 10 corresponds to the RTT $T$ and is due to the periodicity of the model at that scale.

## B. Self-clocking with cross traffic

In the general case, the bottleneck $\mathcal{B}$ may also carry some cross traffic. We assume that the TCP source shares the capacity $C$ of $\mathcal{B}$ with cross traffic in a First-Come First-Served (FCFS) manner. Suppose that the TCP source sends a number $N$ packets with a transmission rate $C_s{>}C$. Let $P_i$ be the $i$'th TCP packet, and $a_i$ be its arrival time at $\mathcal{B}$, with $a_{i+1} - a_i$=$L/C_s$ ($i$=$1 \ldots N-1$). If there are no cross traffic packets arriving between $a_i$ and $a_{i+1}$, the packets $P_i$ and $P_{i+1}$ will depart $\mathcal{B}$ with dispersion $L/C$. Otherwise, the packets $P_i$ and $P_{i+1}$ will depart with a smaller rate than $C$, i.e., a larger dispersion than $L/C$.

Furthermore, suppose that in a particular round-trip the window is $W$ packets and the RTT is $T$ seconds. Due to the presence of cross traffic, the window $W$ can be "segmented", in the general case, in a number $K{\geq}1$ of bursts of rate $C$ and length $B_i$, with $\sum_{i=1}^{K} B_i$=$W$, as shown in Figure 4-c. The off-period between two successive bursts is $I_i$, while the total dispersion of the window, i.e., the time distance between the first and last packets of the window, is $\Delta$. Note that this structure is a two-level ON/OFF pattern. At the lower level, each ON duration consists of a bursts of $B_i$ packets sent at the full capacity of the forward path $C$. At the higher level, the ON duration consists of a cluster of bursts, totally $W$ packets long, with duration $WL/C{\leq}\Delta{<}T$.

To construct a sub-RTT model of TCP packet interarrivals that follows the two-level ON/OFF structure, we need to characterize the distribution of $\Delta$, and of each $B_i$ and $I_i$. Attempting to create a relatively simple model, with as few parameters as possible, we will simply assume that $\Delta$ is constant, the burst lengths $B_i$ are geometric random variables, and the $K$-1 idle times $I_i$ are pseudo-exponential.

Specifically, the burst length $B_i$ follows the geometric distribution when the probability of cross traffic arrivals in any interval of length $a_{i+1} - a_i$=$L/C_s$ is constant, say equal to $p$, independent of $i$. The average burst length then is $\bar{B}$=$1/p$. The sum of the burst lengths during an RTT should be limited by $W$, and so the last burst length $B_K$ during an RTT follows a truncated geometric distribution with maximum length $W$ if $K$=1, and $W$-$(B_1 + \ldots B_{K-1})$ if $K{>}1$. Given $\Delta$ and $K$, the idle times $I_i$ between successive bursts are modeled as pseudo-exponential with parameters ($K$-1, $\Delta$ - $\frac{WL}{C}$) (see §II).

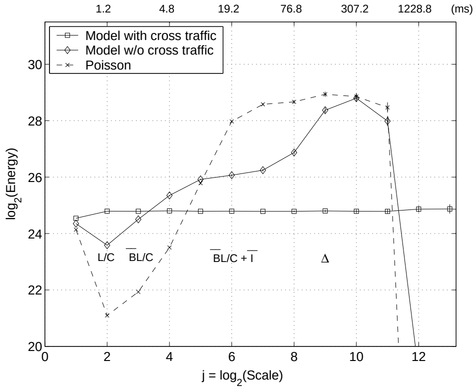


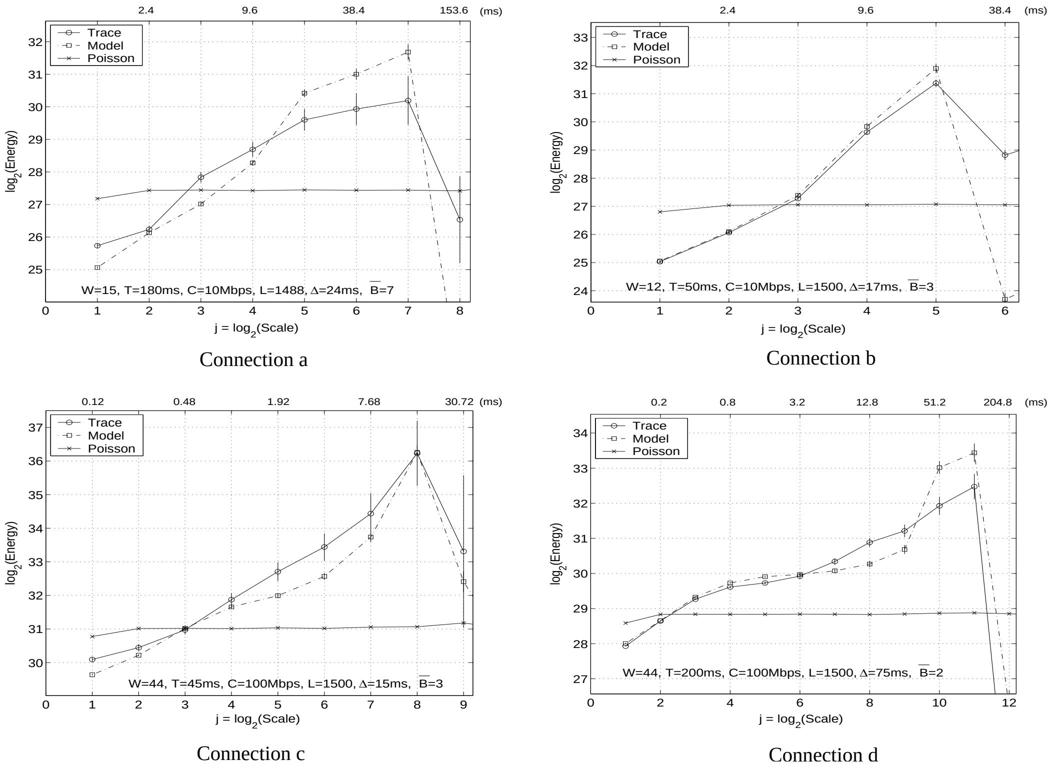
Fig. 6. Energy plot for TCP self-clocking model with cross traffic.

Notice that the previous model involves six parameters: $W$, $T$, $L$, $C$, $\Delta$ and $\bar{B}$; the remaining model variables can be determined from these parameters. Figure 6 shows the energy plot that results from the previous model for $W$=16 packets, $T$=1228ms, $L$=1500 bytes, $C$=10Mbps, $\Delta$=128ms, and $\bar{B}$=2 packets. Note that the energy increases in two regions, corresponding to the two ON/OFF patterns: first, between $L/C$ and $\bar{B}L/C$, and second, between $\bar{B}L/C + \bar{I}$ and $\Delta$. Overall, the process is bursty in almost the entire range of sub-RTT time scales. Figure 6 also shows the energy plot for the corresponding one-level ON/OFF train model, with the same average rate $WL/T$, that would result without cross traffic. Notice that the two models show different degrees of burstiness in different time scales. The one-level ON/OFF model is burstier at the higher end of sub-RTT scales because its packet trains are not segmented into smaller clusters.

## C. Examples of sub-RTT TCP burstiness

To validate the previous model, we compare the energy plots of several long TCP flows with the energy plots that the previous model produces, when parameterized based on the characteristics of the corresponding TCP flows. Specifically, given a packet trace of a long TCP flow, we estimate its RTT $T$ and forward path capacity $C$ as described in §V, while $L$ is simply set to the Maximum Segment Size. Then, we split the flow in successive round-trips of length $T$, resulting in a sequence of windows from which we estimate $W$ as the 75-th percentile window measurement. The average burst length $\bar{B}$ and the total dispersion $\Delta$ are similarly estimated from the trace, based on the median of all the burst and dispersion measurements, respectively.

Figure 7 shows the trace-based and model-based energy plots for four large TCP transfers that we picked randomly from an OC-48 packet trace (more details for the trace are given in §V). The six parameters for each flow are also shown



Connection a

Connection b

Connection c

Connection d

Fig. 7. Trace-based and model-based energy plots for four large TCP transfers

in the graphs. The vertical bars at the energy plot of the TCP traces are confidence intervals for the wavelet-based energy estimates. Clearly, the match between the model and the real traffic is not perfect. One possible reason for the differences between traces and model is that the previous model assumes the same window length in every round-trip, and so it only captures the correlations between packet arrivals within the same round-trip; correlations between different round-trips, however, can also exist. After analyzing many TCP traces, we are convinced that it is very hard to model the correlation structure of TCP in sub-RTT scales in a parsimonious way. Nevertheless, the previous model is a good approximation, as it is able to capture the basic shape of the energy plot of real TCP flows in sub-RTT scales.

## IV. The effects of aggregation

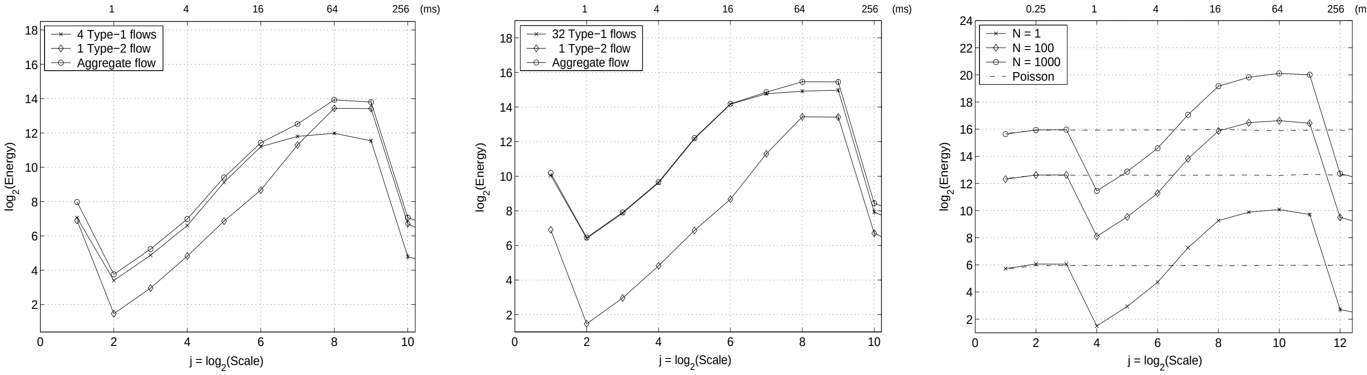
In the previous section, we showed that a single TCP flow can be bursty in sub-RTT scales, due to the strong correlations introduced by self-clocking. What happens, however, when we multiplex $N$ TCP flows in the same link? How does the burstiness of the aggregate flow relate to the burstiness of its constituents? Would the “intermingling” of packets from different flows mitigate the correlations between packets of individual flows?

Consider the *superposition*, or aggregation, of $N$ independent TCP flows. Each flow follows the sub-RTT model of the previous section, determined by the six parameters ($W$, $T$, $L$, $C$, $\Delta$, $\bar{B}$). Suppose we have two types of flows. A Type-1 flow has $W$=16pkts, $T$=256ms, $L$=1250B, $C$=10Mbps, $\Delta$=16ms, $\bar{B}$=16pkts, and so it tends to send the entire window as a single burst of rate $C$. A Type-2 flow has the same $T$, $L$, and $C$, while $W$=64pkts, $\Delta$=64ms, $\bar{B}$=64pkts, and so its average rate is four times larger than that of a Type-1 flow.

Suppose that we aggregate flows with different sub-RTT burstiness, but with the same average rate. That would be the case, for instance, if we aggregate four Type-1 flows with one Type-2 flow. The energy plots of the aggregate stream and of the constituent flows are shown in Figure 8-a. Note that the energy plot of the aggregate follows the energy plot of the four Type-1 flows up to scale 7, and of the Type-2 flow in larger scales. In general, when we aggregate flows with different burstiness but equal rates, *the burstiness of the aggregate flow at a given scale is determined by the the constituent flows with the maximum burstiness at that scale*.

On the other hand, it is hard to predict the burstiness of the aggregate flow when the constituents have different rates. A general observation, however, is that *flows with a minor contribution in the aggregate throughput also have a minor contribution in the burstiness of the aggregate stream*. For instance, suppose that we aggregate 32 Type-1 flows with one Type-2 flow. The latter accounts for only 11.1% of the aggregate flow’s throughput. The energy plots of the aggregate flow and of its constituents’ are shown in Figure 8-b. Notice that the energy plot of the aggregate follows the energy plot of the Type-1 flows, while the presence of the Type-2 flow has a minor impact on the burstiness of the aggregate.

The last observation has an important practical implication: *when analyzing the burstiness of aggregate traffic, we can ignore constituent flows that only account for a small part of the whole, and focus on the larger flows only.* Several



(a) Aggregation of flows with different burstiness but equal rates.

(b) Aggregation of flows with different rates and burstiness

(c) Aggregation of N Type−1 flows for different values of N.

Fig. 8. Energy plots for aggregation experiments

measurement studies have shown that typically 80-90% of the traffic are carried by a small fraction (about 10%) of the flows. Consequently, it is reasonable to expect that the burstiness of these larger flows determines the burstiness of the aggregate traffic.

The major question, however, is what happens to the burstiness of the aggregate stream as we increase $N$? Figure 8-c shows the energy plots of the aggregate of $N$ Type-1 flows for $N$=1, 100, and 1000. Clearly the energy increases with $N$, because the average rate of the aggregate stream increases. More importantly, however, note that *the energy plot maintains the same shape as that of a single Type-1 flow, independent of $N$, i.e., only the magnitude of the energy changes*.

The fact that the energy plot of the aggregate does not gradually becomes closer to a horizontal line implies that the aggregate flow does *not* tend to the Poisson process, and that it maintains the same correlation structure with any of its constituents. This may appear at first as contradictory to a classical result from the theory of point processes, according to which the superposition of $N$ independent point processes converges to the Poisson process as $N$ increases [26]. There is *no* actual contradiction however. The previous result assumes that the rate of each constituent flow is $\lambda/N$, so that the rate of the aggregate flow remains the same, equal to $\lambda$, independent of $N$. In other words, each constituent flow becomes gradually “sparser” as $N$ increases. In packet networks, on the other hand, the aggregation of $N$ flows with average rate $\lambda$ creates a stream of rate $N\lambda$, which is then serviced by a link with capacity at least $N\lambda$. Consequently, in packet networks, the interarrivals of individual flows are not scaled by a factor of $N$ when the flows are multiplexed in the same link.

Actually, the previous issue has been discussed in an earlier paper by Sriram and Whitt [33]. They showed that, if the rate of the constituent flows does not decrease with $N$, the interarrivals of the aggregate flow tend to exponential, but they do not loose their correlation structure. That is easy to show for $N$ homogeneous flows. Let $X_{i,j}$ be the counting process for flow $j$ at a given scale $T$, i.e., the number of bytes from flow $j$ in the $i$'th time interval of length $T$, and $Y_i = \sum_{j=1}^{N} X_{i,j}$, the counting process for the aggregate flow. We assume that $X_{i,j}$ is stationary in $i$, and independent and homogeneous in $j$. Let $r(k)$ be the autocorrelation of the aggregate at lag $k$, and $r_j(k)$ be the autocorrelation of flow $j$ at the same lag. Then

$$r(k) = \frac{\text{Cov}(Y_i, Y_{i+k})}{\text{Var}(Y)} = \frac{\sum_{j=1}^{N} \text{Cov}(X_{i,j}, X_{i+k,j})}{\sum_{j=1}^{N} \text{Var}(X_j)}$$
$$= \frac{\text{Cov}(X_{i,j}, X_{i+k,j})}{\text{Var}(X_j)} = r_j(k) \qquad (5)$$

So, the aggregate flow has the same autocorrelation function with any of its constituents, independent of $N$.

We make a final remark on the effect of aggregation in the burstiness of network traffic. Even though, as previously shown, aggregation does not lead to uncorrelated arrivals, it does improve queueing performance in the following sense. As $N$ increases, the marginal distribution of the aggregate flow tends to Gaussian. Since both the mean and variance of that distribution increase proportionally to $N$, the coefficient of variation decreases with the square-root of $N$. So, the traffic appears to be “smoother” simply because the magnitude of the traffic variations at any given time scale decrease in magnitude relative to the average traffic rate, as $N$ increases. We emphasize, however, that this is a basic property of statistical multiplexing for independent flows, and it does not relate to the correlation structure of the aggregate traffic, nor it implies that the traffic tends to the Poisson model.

## V. Case study: the burstiness of an OC-48 trace

In this section, we apply the insight of the previous two sections in the analysis of an OC-48 trace from a major Internet2 backbone link. Starting from a large trace, with almost half a million flows, our goal is to identify the flows that determine the short scale burstiness of the entire trace. Our (unidirectional) trace was collected at the OC-48 link that connects the Abilene routers from Cleveland (CLEV) to Indianapolis (IPLS). Even though the trace, which is publicly available through NLANR-MOAT [34], covers two hours (9:00-11:00 on 8/14/2002), we analyze here a 90-sec segment from 10:08:30 to 10:10:00.

We start from the original trace, say $S_{org}$, and gradually form an increasingly narrower subset of flows that are responsible for the short scale burstiness of the entire trace. This

| Subset | GBytes | TCP flows | Percentage of bytes | Percentage of packets |
|---|---|---|---|---|
| $S_{org}$ | 4.37 | 458669 | 100 | 100 |
| $S_{tcp}$ | 4.23 | 458669 | 96.8 | 93.6 |
| $S_{\bar{tcp}}$ | 0.14 | N/A | 3.2 | 6.4 |
| $S_{rtt}$ | 2.41 | 40885 | 55.1 | 46.6 |
| $S_{\bar{rtt}}$ | 1.83 | 417784 | 41.7 | 47.0 |
| $S_{rtt52}$ | 0.75 | 9041 | 17.1 | 13.2 |
| $S_{bdp}$ | 2.25 | 10484 | 51.5 | 27.0 |
| $S_{lrg}$ | 2.22 | 3207 | 50.9 | 25.3 |
| $S_{sml}$ | 0.03 | 7277 | 0.6 | 1.7 |
| $S_{lr\Theta}$ | 2.12 | 3123 | 48.6 | 24.2 |
| $S_{sm\Theta}$ | 0.10 | 84 | 2.3 | 1.1 |

TABLE I

SUBSETS OF THE IPLS OC-48 TRACE

investigative work, which is similar to gradually reducing a set of crime suspects through additional evidence, leads us eventually to a relatively small set of flows (about 3,100 flows, out of 460,000 in the original trace) that dominate the short scale burstiness of the entire trace. As expected from §III and §IV, we find that this set includes *bulk TCP flows that have a large bandwidth-delay product relative to their window size*. Table I shows the notation, and a few statistics, for the various sets of flows that we consider in the remaining of the section.

First, Figure 9-a shows the energy plot of the original trace $S_{org}$, together with the energy plot of a Poisson process with the same average rate. Note that the trace shows strong burstiness in short time scales, up to about 200ms. In longer scales, the trace exhibits a range of linearly increasing energy due to LRD effects, which is typical of Internet traffic (see [9] or [12] for similar examples of this "bi-scaling" behavior in short vs long time scales). The boundary between short and long scales is the dramatic energy drop around scale 10. We will relate that scale with the RTT of the TCP flows in the trace shortly.

Figure 9-a also shows the energy plot of the subset $S_{tcp}$, that includes only TCP traffic. Since TCP accounts for 97% of the byte-traffic in this trace, it should not be surprising that the energy plot of $S_{tcp}$ is basically the same with that of $S_{org}$. Even if the rest of the traffic, denoted by $S_{\bar{tcp}}$, had some interesting burstiness characteristics, it would not be able to affect the energy plot of the entire trace due to its small volume.

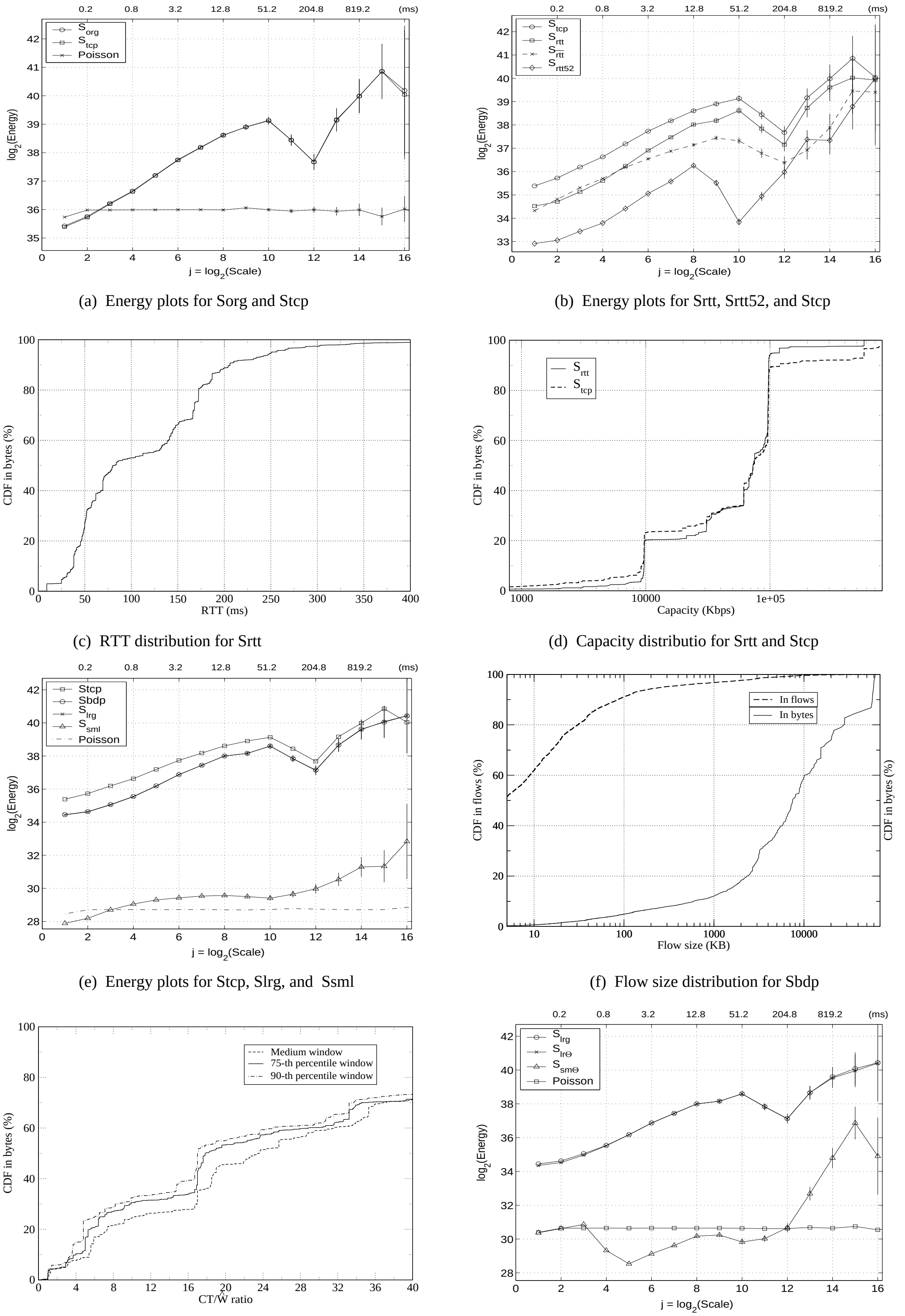
Next, we identify the extent of the short scale burstiness, and relate it to the RTT distribution of the TCP flows. To do so, however, we first need to know the RTT of each TCP flow in the trace. That is hard to do, especially for a unidirectional trace. The estimation technique proposed in [35] can provide a single RTT measurement per connection for a significant fraction of the TCP traffic in a trace. Using that technique, we formed a new subset, $S_{rtt}$, which includes all TCP flows for which we have RTT estimates; TCP flows without an RTT estimate belong in $S_{\bar{rtt}}$. Even though the number of flows in $S_{\bar{rtt}}$ is much larger than in $S_{rtt}$, the latter includes more than 50% of the bytes in the trace. Figure 9-b shows the energy plots of the two subsets, which are quite similar in shape and magnitude. Thus, we can view $S_{rtt}$ as an unbiased sample of the $S_{tcp}$ trace, at least in terms of burstiness.

The estimated RTT distribution for $S_{rtt}$ is shown in Figure 9-c, plotted in terms of bytes rather than flows. Note that most of the traffic in that subset is carried by flows with RTT in the range 25-250ms. The weighted average of the flow RTTs, with each RTT measurement weighted by the fraction of bytes in the corresponding flow, is about 117ms. Notice that the dip in the energy plot of Figure 9-a occurs at scale 12 (about 200ms), which is close to the (weighted) average RTT of the trace (about 120ms). In other words, the major drop in the energy plot of an aggregate trace is located at about the same time scale with the RTT of the dominant flows in the trace. This should not be surprising. As explained in §II, a periodicity in the traffic process causes a drop in the energy plot at the time scale that corresponds to the period. The RTT of a TCP flow, however, represents a natural periodicity in its traffic process as long as the flow's window does not vary significantly from round to round.

To further examine the previous conjecture, we form another trace subset, $S_{rtt52}$, of all flows in $S_{rtt}$ for which the RTT is less than 52ms. The weighted average RTT of $S_{rtt52}$ is 39ms, and its energy plot is shown in Figure 9-b. Note that the energy plot dip has now moved to scale 10 (51.2ms), which corresponds to the same scale with the previous weighted average. To summarize, *the dip that is commonly seen in energy plots of Internet traffic occurs at roughly the same time scale with the RTT of the dominant TCP flows, and it provides an easy way to identify the extent of sub-RTT burstiness.*

To estimate the bandwidth-delay product of a TCP flow, we need to know its forward-path capacity $C$, as defined in §III, together with its RTT. That capacity of a TCP flow can be estimated using the packet-pair technique, given that TCP sends many packet pairs due to Delayed-ACK. Using a passive capacity estimation technique that we developed [36], we can estimate the capacity of about 90% of the bytes in $S_{tcp}$, and of the entire $S_{rtt}$. Figure 9-d shows the capacity distribution for the two sets. Notice that about 40% of the bytes belong to high-capacity flows ($C \approx$100Mbps), while about 90% of the bytes are generated from flows with $C \geq$10Mbps. These relatively high capacity values may be due to the fact that the trace captures Internet2, rather than commercial Internet, traffic. The set of TCP flows for which we have both an RTT and a capacity estimate is denoted by $S_{bdp}$. Its energy plot is shown in Figure 9-e. Note that $S_{bdp}$ has practically the same energy plot with $S_{tcp}$, and thus with $S_{org}$, implying that it is an unbiased sample of the original trace in terms of sub-RTT burstiness.

We next focus on the link between flow size and sub-RTT burstiness. The conjecture, based on the insight from §IV, is that it is the large flows that determine the shape of the aggregate energy plot. Figure 9-f shows the cumulative flow size distribution of $S_{bdp}$ in terms of both bytes and flows. Note that 70% of the flows are shorter than 15KB, but they account for less than 2% of the aggregate traffic. Such a heavy-tailed flow size distribution is typical of Internet traffic, and it agrees with the general classification between "elephants" and "mice". We split the flows of $S_{bdp}$ into the set $S_{lrg}$, which includes all flows larger than 15KB, and $S_{sml}$, which includes



(a) Energy plots for Sorg and Stcp

(b) Energy plots for Srtt, Srtt52, and Stcp

(c) RTT distribution for Srtt

(d) Capacity distributio for Srtt and Stcp

(e) Energy plots for Stcp, Slrg, and Ssml

(f) Flow size distribution for Sbdp

(g) Ratio distribution for Slrg

(h) Energy plots for SlrΘand Ssm Θ

Fig. 9. Analysis of OC-48 trace

the remaining, small flows.

Figure 9-e shows the energy plots of the $S_{lrg}$ and $S_{sml}$ sets, together with the energy plots of the $S_{bdp}$ and $S_{tcp}$ sets. First, note that the energy plot of $S_{lrg}$ completely overlaps with that of $S_{bdp}$, confirming our previous conjecture that large flows determine the shape of the energy plot for the entire trace. That is not surprising given that $S_{sml}$ accounts for only 0.6% of the bytes in the original trace. A second interesting observation is that the energy plot of $S_{sml}$ appears to be almost horizontal beyong scale 5. That may be interpreted by someone as an indication of a Poisson-like process. That is not the case however. Comparing the energy plot of $S_{sml}$ with that of a Poisson process of the same average rate shows that $S_{sml}$ is bursty in almost the entire sub-RTT range of time scales.

Now that we have identified the large TCP flows as "primary suspects" for sub-RTT burstiness, and since we have an estimate of the bandwidth-delay product for about 51% of the bytes ($S_{bdp}$) in the trace, we can return to our original conjecture: large TCP flows cause sub-RTT burstiness if their bandwidth-delay product $CT$ is large relative to their window size $WL$.

To examine this conjecture, we estimate the number of bytes $(WL)_i$ in each round-trip $i$, for every TCP flow in $S_{lrg}$. Note that $S_{lrg}$ contains only data flows since we do not have capacity estimates for ACK flows. Obviously, the window measurements $(WL)_i$ vary with $i$, as the TCP congestion window changes. In the following, we use a certain percentile, denoted by $\hat{W}$, of a flow's window size distribution to calculate the ratio $\Theta$=$CT/\hat{W}$. $\Theta$ is the ratio of the bandwidth-delay product to a "typical" window size $\hat{W}$ for each flow in $S_{lrg}$. Figure 9-g shows the distribution of $\Theta$ for three percentiles $\hat{W}$: 50-th, 75-th, and 90-th. Note that the distribution is not so sensitive to the exact definition $\hat{W}$, especially in lower values of $\Theta$. In the following, we define $\hat{W}$ based on the 75-th percentile.

An important observation in Figure 9-g is that $\Theta$ is quite larger than 1.0 for most of the traffic in $S_{lrg}$. Specifically, more than 90% of the bytes in $S_{lrg}$ have $\Theta$>4.0, while as little as 5% of the bytes have $\Theta$<2.0. We further split $S_{lrg}$ into $S_{lr\Theta}$ and $S_{sm\Theta}$, where the former includes the TCP flows with $\Theta$>2.0, while the rest of $S_{lrg}$ forms $S_{sm\Theta}$. The energy plots of the two new subsets are shown in Figure 9-h. First, note that the energy plot of $S_{lr\Theta}$ completely overlaps with $S_{lrg}$, verifying our main conjecture that *flows with large values of* $\Theta$ *determine the sub-RTT burstiness of the entire trace*. Second, perhaps more surprisingly, the energy plot of $S_{sm\Theta}$ is below the energy of a Poisson process with the same rate as that subset, i.e., *TCP flows with small values of* $\Theta$*, close to 1.0, are smooth and they do not contribute to the sub-RTT burstiness of the aggregate traffic.*

We can summarize the case-study of this section as follows: the short time scale burstiness of this OC-48 trace extends up to the weighted average RTT of the TCP flows. The sub-RTT burstiness is due to large TCP flows (more than 15KB) that have at least twice as large bandwidth-delay product relative to their typical window size. Non-TCP traffic, or small TCP flows (less than 15KB), do not contribute to the burstiness of the aggregate traffic due to their very small volume. Finally, large flows with roughly equal window sizes and bandwidth-delay products create Poisson-like, or even smoother, traffic.

```
for a round-trip with window $W_i$ and RTT $T$ {
    $X$ = $X_{min}$;
    $n$ = $max(1, \lfloor \frac{T}{\lceil \frac{W}{X} \rceil T_c} \rfloor)$;

    while ($T < \lceil \frac{W}{X} \rceil n T_c$  &&  $X < W$) {
        $X$++;
        $n$ = $max(1, \lfloor \frac{T}{\lceil \frac{W}{X} \rceil T_c} \rfloor)$;
    }
    return ($n$, $X$);
}
```

Fig. 10. Pseudo-code for calculating $X$ and $n$.

## VI. The smoothing effect of TCP pacing

Sections III and V showed that self-clocking is largely responsible for the burstiness of both individual TCP flows and aggregate Internet traffic. The basic problem with self-clocking is that, under certain conditions, it causes TCP to send its entire window as a long burst, or as a cluster of bursts, instead of distributing that window's packets during the corresponding round-trip [16]. One way to remove, or at least reduce, the burstiness of a TCP flow is to perform *pacing* at the sender [37], [38]. With pacing, TCP sends packets periodically during the corresponding round-trip, instead of being driven by the arrival timing of ACKs.

Suppose that in a particular round-trip the TCP sender has a window of length $W$ packets[3], and that the RTT is $T$. According to *ideal pacing*, the sender should send packets periodically, every $T/W$ time units, i.e., at a rate that is equal to the flow's average throughput $W/T$ in that round-trip. We refer to this scheme as "ideal" because it would require scheduling transmission events in arbitrary intervals. Such a timing facility would be impractical, or it would introduce a large overhead at the sender, especially for flows with large $W$ and small $T$. For example, if $T$=15ms and $W$=100 packets, the sending host would have to generate a timeout, to schedule a packet transmission, in every 150$\mu$s. In practice, commodity operating systems typically provide a minimum timer of either 10ms, or 1ms in the best case.

TCP pacing will not be practical unless if we consider the presence of a *minimum pacing timeout* $T_c$, with values in the range of 1-10ms. Given that the TCP sender can only schedule packet departures every $nT_c$ time units, where $n$ is a positive integer, it may be required to send multiple packets back-to-back when $T/T_c$<$W$. An algorithm that computes the minimum burst length $X$, and the corresponding value of $n$, is given in Figure 10. $X_{min}$ is the minimum burst size (for instance, 1-2 packets) imposed. Note that the algorithm finds the minimum value of $X$ that allows the transmission of $W$ packets during $T$ time units given that it is only possible to send packets every $nT_c$ time units.

Let us now examine the burstiness reduction with ideal pacing as well as with practical pacing for $T_c$=10ms or 1ms

[3] For simplicity we assume here that the window is measured in packets instead of bytes.

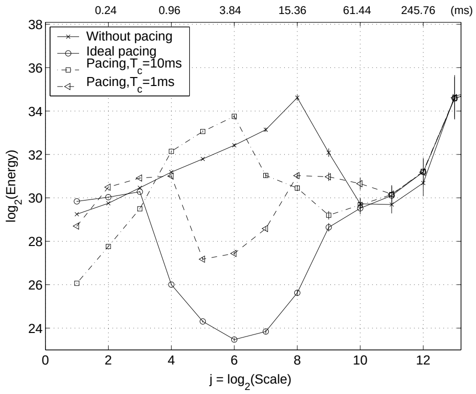


Fig. 11. The effect of ideal and practical pacing on a TCP flow with $C$=100Mbps, $T$=48ms, $L$=1500B, $W_{75-th}$=45pkts, and $W_{50-th}$=44pkts

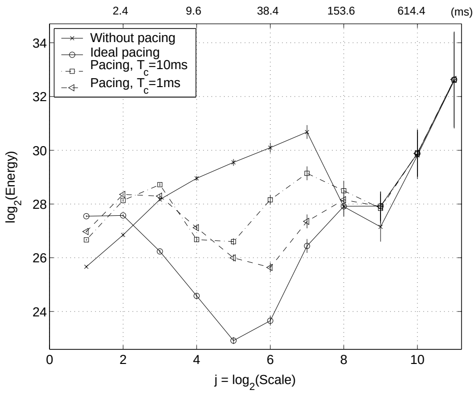


Fig. 12. The effect of ideal and practical pacing on a TCP flow with $C$=10Mbps, $T$=151ms, $L$=1500B, $W_{75-th}$=24pkts, and $W_{50-th}$=15pkts

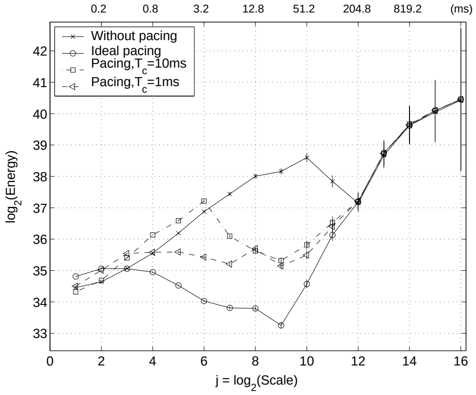


Fig. 13. The effect of ideal and practical pacing on the $S_{bdp}$ set of §V

and $X_{min}$=2 packets. We first show the impact of pacing on individual TCP flows from the OC-48 trace of §V. The RTT $T$ and the sequence of window lengths $\{W_i\}$ for each round-trip $i$ are estimated as explained in §V. Second, we show the impact of pacing on aggregate traffic, after performing pacing on each of the constituent flows.

Figure 11 shows four energy plots for a TCP flow that lasted 26.4 seconds and generated 13450 data packets. The RTT of the flow is approximately $T$=48ms, while its 75th percentile window length was 45 packets. One of the curves in Figure 11 is the energy plot of the original trace, without pacing. Note that the flow has strong sub-RTT burstiness (scales 2-8). The energy level of the corresponding Poisson process is 30.1. With ideal pacing, the traffic becomes extremely smooth, as we would expect, due to the periodic nature of packet departures in every round-trip. Also note that pacing does not affect the burstiness of the trace in time scales that extend beyond the RTT (scale 10 and higher). The two curves that correspond to practical pacing show clearly that the minimum pacing timer $T_c$ has a major impact on the smoothing effectiveness of pacing. The 10ms timer is unable to reduce the burstiness of the flow up to scale 6. The reason is that, with $T_c$=10ms, the burst length $X$ is often as large as more than 10 packets. The 1ms timer, on the other hand, is much more effective in reducing the burstiness throughout the sub-RTT scales. Even though it is not as effective as ideal pacing, the 1ms timer manages to reduce burstiness at the level of the corresponding Poisson process, or even less than that.

Figure 12 shows similar results for a different TCP flow with significantly larger RTT ($T$=151ms) and with smaller windows (the 75-th percentile window is 24 packets). The fact that the ratio $T/W$ is larger, compared to the flow of Figure 11, makes it easier for the practical pacing schemes to schedule short burst lengths (up to 3 or 4 packets). The 1ms pacing timer is, of course, still more effective than the 10ms timer, and it manages to reduce the burstiness of the flow to the level of the corresponding Poisson process (energy 27.7) in the sub-RTT scales.

If pacing is effective in reducing the burstiness of individual TCP flows, how does it affect the burstiness of an aggregate of paced TCP flows? We performed pacing, both ideal and practical, on every TCP flow in the $S_{bdp}$ set of §V. Figure 13 shows the resulting energy plots. Ideal pacing would be able to make the aggregate flow smooth throughout the sub-RTT scales, up to scale 10 (50ms). That is not the case with a minimum pacing timer of $T_c$=10ms, mostly due to remaining burstiness in very small time scales (up to 3-4ms). On the other hand, the 1ms pacing timer would be effective in reducing the energy of TCP traffic in sub-RTT scales to the level a Poisson process with the same average rate.

## VII. Queueing performance

In this section, we examine the impact of sub-RTT burstiness on queueing performance, in terms of the queue size distribution and packet loss rate. We also evaluate the improvement in queueing performance that results from the TCP pacing schemes of the last section, in various load and buffer size operating points.

Our evaluation is based on trace-driven queueing simulations. Specifically, we simulate the queue of an OC-3 output link that is loaded with traffic from two OC-48 links. The traffic in each OC-48 ingress link is generated based on two non-overlapping 90-second segments (09:58:30-10:00:00, and 10:08:30-10:10:00) of the two-hour OC-48 trace that we mentioned in §V. The two segments include only flows for which we have both RTT and capacity estimates, and they are similar in terms of their average rate, energy plot, flow size distribution, and RTT and capacity distributions. We note that such a trace-driven simulation is based on realistic Internet

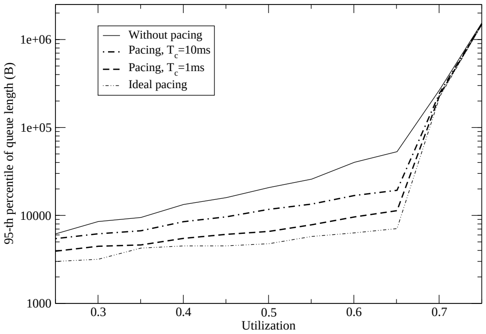


Fig. 14. 95-th percentile of the queue size (in bytes) as a function of the utilization ($B$=10MB)

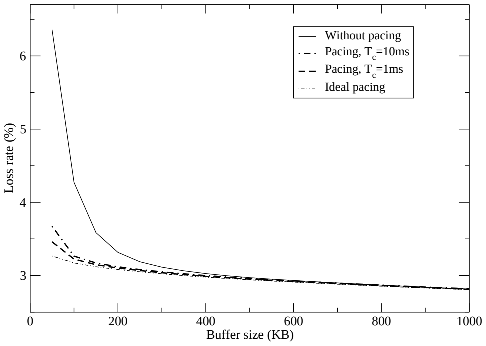


Fig. 15. Loss rate as a function of the buffer size $B$ ($\rho = 0.95$)

traffic, but it does not capture the closed-loop nature of TCP traffic, since simulated losses do not lead to an offered load reduction.

To simulate a particular utilization $\rho_i$, we have to adjust the average rate that is fed into the OC-3 link to $\rho_i \times$155Mbps. To do so, we select a random set of flows from each of the input traces that produces approximately that average rate. This process is more complex than similar trace-driven simulations in previous related work, but we believe it is more realistic, as we do not artificially modify the packet interarrivals of the input traffic or the capacity of the output link in order to achieve a certain utilization. We note that the average rate of each input trace is roughly 200Mbps, meaning that the previous technique can produce an offered load of up to 400Mbps, sufficient to saturate the OC-3 link.

We perform simulations with four instances of the input traffic. The first is with the original traffic, and so it includes sub-RTT burstiness. The second simulation is with the ideally paced traffic, in which each of the constituent TCP flows has been ideally paced as described in §VI. The third and fourth simulations are, again with paced TCP flows, but this time with a 10ms and 1ms minimum pacing timer, respectively. It is important to note that pacing, either ideal or practical, does *not* change the burstiness of a TCP flow in scales that extend beyond its RTT.

The outcome of each simulation is either the 95-th percentile of the queue size distribution, or the packet loss rate. The buffer size of the OC-3 queue is $B$ bytes, and it is organized in terms of variable-sized buffers (as opposed to packet-based buffers). A packet is dropped when there is no available buffer space at the OC-3 queue. Each simulation is repeated 10 times with a different random set of input flows. We report the average of those results.

Figure 14 shows the results for the 95-th percentile of the queue size distribution. As long as the utilization is between 30% to 70%, i.e., in moderate load conditions, ideal pacing would reduce the tail of the queue size distribution by about an order of magnitude. The reason is that pacing removes the sub-RTT burstiness of the traffic. Very similar benefits would result from pacing with the 1ms minimum timer, but not with the 10ms timer.

The most striking, however, result of Figure 14 is that when the utilization is 70% or higher, i.e., in heavy-load conditions, the 95-th percentile of the queue size increases exponentially, and the benefit of pacing becomes negligible. The reason is that, in heavy-load conditions, the LRD properties of the traffic become much more important than the sub-RTT burstiness. Even though pacing is effective in removing the latter, it cannot make the traffic smoother in scales that extend beyond the RTT of a TCP flow. The reason that LRD dominates queueing performance in heavy-load conditions is that variability in large time scales causes an increased utilization over significant time periods. When the long-term average utilization is already high (70% or more), any further increases can significantly overload the queue. This effect has been previously studied in [19].

The buffer size in the simulations of Figure 14 was $B$=10MB, which corresponds to 500ms buffering for an OC-3 link. This is a relatively large buffer size, typical for an overbuffered router interface today. What happens, however, when $B$ is significantly smaller? Figure 15 shows the resulting loss rate as a function of $B$ for a heavy-load operating point ($\rho$=95%). An interesting effect takes place when the buffer size is shorter than 200KB: *the loss rate with pacing, i.e., without short time scale burstiness, is significantly lower than the loss rate in the original trace*. Note that this may seem contradictory to the results of [19], since that work showed that short-range dependencies (which include sub-RTT TCP burstiness) are not significant in heavy-load conditions. The key point, however, is that [19] considered a lossless queue, and so, practically, a very large buffer size $B$. With relatively small buffers, on the other hand, the impact of long-range dependency decreases [4], and so the effect of short-range burstiness becomes more noticeable.

To further demonstrate the previous point, Figure 16 shows

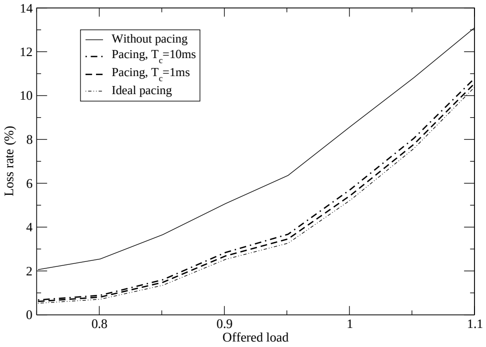


Fig. 16. Loss rate as a function of the offered load for an underbuffered link ($B$=50KB)

the loss rate as a function of the offered load for a link with $B$=50KB. Note that pacing causes an important reduction in the loss rate not only in moderate-load conditions, as would be the case with an wellbuffered link, but also in heavy-load conditions. We note that it is not rare for Internet links to be underbuffered, either due to poor provisioning, or because the links are optimized for real-time traffic.

## VIII. Conclusions

The cause of short scale burstiness in Internet traffic has been highly debated in the last few years. We showed that TCP self-clocking, coupled with queueing in the bottleneck of the connection's forward path, can create ON/OFF interarrival structures, and thus, strong correlations and burstiness. Such structures are generated when the bandwidth-delay product is large relative to the flow's window. Aggregating many TCP flows in the same link does not produce less correlated traffic, as previously argued. Instead, the observed multiplexing gains are due to a smoother marginal distribution. The analysis of an OC-48 trace confirmed that the burstiness of aggregate traffic in short time scales extends up to the RTT of the dominant TCP flows, and it is due to large TCP flows that have a high bandwidth-delay product over window ratio. An effective way to reduce the sub-RTT burstiness of Internet traffic is to perform pacing at the sources, especially if the minimum pacing timer can be in the order of 1ms. Finally, in the presence of LRD effects, sub-RTT burstiness is mostly important in moderate load conditions and/or in relatively small link buffers.

## IX. Acknowledgement

We greatly appreciate the availability of traces from the NLANR MOAT project. The NLANR MOAT project is supported by National Science Foundation Cooperative agreement nos. ANI-0129677 (2002) and ANI-9807479 (1998).

## References

[1] W. E. Leland, M. S. Taqqu, W. Willinger, and D. V. Wilson, "On the Self-Similar Nature of Ethernet Traffic (Extended Version)," *IEEE/ACM Transactions on Networking*, vol. 2, pp. 1–15, Feb. 1994.

[2] K. Park and W. Willinger (editors), *Self-Similar Network Traffic and Performance Evaluation*. John Willey, 2000.

[3] A. Erramilli, O. Narayan, and W. Willinger, "Experimental Queueing Analysis with Long-Range Dependent Packet Traffic," *IEEE/ACM Transactions on Networking*, vol. 4, pp. 209–223, Apr. 1996.

[4] M. Grossglauser and J.-C. Bolot, "On the Relevance of Long-Range Dependence in Network Traffic," *IEEE/ACM Transactions on Networking*, vol. 7, no. 5, pp. 629–640, 1999.

[5] 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," in *Proceedings of ACM SIGCOMM*, pp. 100–113, Sept. 1995.

[6] M. E. Crovella and A. Bestavros, "Self-Similarity in World Wide Web Traffic: Evidence and Possible Causes," *IEEE/ACM Transactions on Networking*, vol. 5, pp. 835–846, Dec. 1999.

[7] L. Guo, M. Crovella, and I. Matta, "Corrections to "how does tcp generate pseudo-self-similarity?"," *ACM SIGCOMM Computer Communication Review*, vol. 32, no. 2, pp. 30–30, 2002.

[8] D. R. Figueiredo, B. Liu, and V. M. adn D. Towsley, "On the Autocorrelation Structure of TCP Traffic," *Computer Networks Journal*, 2002.

[9] A. Feldmann, A.C.Gilbert, W.Willinger, and T. G. Kurtz, "The Changing Nature of Network Traffic: Scaling Phenomena," *ACM Computer Communication Review*, Apr. 1998.

[10] A. Feldmann, A.C.Gilbert, and W.Willinger, "Data Networks as Cascades: Investigating the Multifractal Nature of the Internet WAN Traffic," in *Proceedings of ACM SIGCOMM*, 1998.

[11] R. Riedi, M. S. Crouse, V. Ribeiro, and R. G. Baraniuk, "A Multifractal Wavelet Model with Application to Network Traffic," *IEEE Transactions on Information Theory*, vol. 45, pp. 992–1019, Apr. 1999.

[12] Z.-L. Zhang, V. Ribeiro, S. Moon, and C. Diot, "Small-Time Scaling behaviors of Internet backbone traffic: An Empirical Study," in *Proceedings of IEEE INFOCOM*, Apr. 2003.

[13] T. D. Dang, S. Molnar, and I. Maricza, "Queueing Performance Estimation for General Multifractal Traffic," *International Journal of Communication Systems*, vol. 16, no. 2, pp. 117–136, 2003.

[14] A. Feldmann, A.C.Gilbert, P. Huang, and W.Willinger, "Dynamics of IP Traffic: A Study of the Role of Variability and The Impact of Control," in *Proceedings of ACM SIGCOMM*, 1999.

[15] L. Zhang, S. Shenker, and D. D. Clark, "Observations on the Dynamics of a Congestion Control Algorithm," in *Proceedings of ACM SIGCOMM*, Sept. 1991.

[16] J. C. Mogul, "Observing TCP dynamics in real networks," in *Proceedings of ACM SIGCOMM*, Aug. 1992.

[17] N. Hohn, D. Veitch, and P. Abry, "Does fractal scaling at the IP level depend on TCP flow arrival processes?," in *Proceedings Internet Measurement Workshop (IMW)*, Nov. 2002.

[18] N. Hohn, D. Veitch, and P. Abry, "Cluster Processes, a Natural Language for Network Traffic," *IEEE Transactions on Signal Processing, special issue on "Signal Processing in Networking"*, 2003. Accepted for publication.

[19] A. Erramilli, O. Narayan, A. L. Neidhardt, and I. Saniee, "Performance Impacts of Multi-Scaling in Wide-Area TCP/IP Traffic," in *Proceedings of IEEE INFOCOM*, Apr. 2000.

[20] R. R. S. Sarvotham and R. Baraniuk, "Connection-level Analysis and Modeling of Network Traffic," in *Proceedings of ACM SIGCOMM Internet Measurement Workshop*, Nov. 2001.

[21] H. Jiang and C. Dovrolis, "Source-Level Packet Bursts: Causes and Effects," in *Proceedings Internet Measurement Conference (IMC)*, Oct. 2003.

[22] M. Garetto and D. Towsley, "Modeling, Simulation and Measurements of Queueing Delay under Long-Tail Internet Traffic," in *Proceedings of ACM SIGMETRICS*, June 2003.

[23] R. Morris and D. Lin, "Variance of Aggregated Web Traffic," in *Proceedings of IEEE INFOCOM*, Apr. 2000.

[24] J. W. X. Tian and C. Ji, "A Unified Framework for Understanding Network Traffic Using Independent Wavelet Models," in *Proceedings of IEEE INFOCOM*, June 2002.

[25] M. F. T. Karagiannis, M. Molle and A. Broido, "A Nonstationary Poisson View of Internet Traffic," in *Proceedings of IEEE INFOCOM*, Mar. 2004.

[26] D. R. Cox and V. Isham, *Point Processes*. Chapman and Hall, London, 1980.

[27] D. L. J. Cao, W. S. Cleveland and D. X. Sun, "Internet traffic tends to poisson and independent as the load increases," in *Nonlinear Estimation and Classification* (B. Y. C. Holmes, D. Denison and B. Mallick, eds.), Springer, 2002.

[28] P. Abry and D. Veitch, "Wavelet Analysis of Long-Range Dependent Traffic," *IEEE Transactions on Information Theory*, vol. 44, pp. 2–15, Jan. 1998.

[29] D. Veitch, "Code for the Estimation of Scaling Exponents." http://www.cubinlab.ee.mu.oz.au/ darryl/secondorder_code.html, July 2001.

[30] D. J. Daley and D. Vere-Jones, *An Introduction to the Theory of Point Processes*. Springer-Verlag, 2002.

[31] R. Jain and S. A. Routhier, "Packet Trains - Measurements and a New Model for Computer Network Traffic," *IEEE Journal on Selected Areas in Communications*, vol. 4, pp. 986–994, Sept. 1986.

[32] V. Jacobson, "Congestion Avoidance and Control," in *Proceedings of ACM SIGCOMM*, pp. 314–329, Sept. 1988.

[33] K. Sriram and W. Whitt, "Characterizing Superposition Arrival Processes in Packet Multiplexers for Voice and Data," *IEEE Journal on Selected Areas in Communications*, vol. 4, no. 6, pp. 833–846, 1986.

[34] NLANR MOAT, "Passive Measurement and Analysis." http://pma.nlanr.net/PMA/, Dec. 2003.

[35] H. Jiang and C. Dovrolis, "Passive Estimation of TCP Round-Trip Times," *ACM Computer Communication Review (CCR)*, Aug. 2002.

[36] H. Jiang and C. Dovrolis, "The effect of flow capacities on the burstiness of aggregated traffic," in *Proceedings Passive and Active Measurements (PAM) workshop*, Apr. 2004. Accepted for publication.

[37] A. Amit, S. Savage, and T. Anderson, "Understanding the Performance of TCP Pacing," in *Proceedings of IEEE INFOCOM*, Apr. 2000.

[38] D. R. J. Kulik, R. Coulter and C. Partridge, “Paced TCP for High Delay-Bandwidth Networks,” in *Proceedings IEEE GLOBECOM*, Dec. 1999.