

Why is the Internet Traffic Bursty in Short Time Scales? *

Hao Jiang, Constantinos Dovrolis
College of Computing, Georgia Tech
{hjiang,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 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. This effect is more important 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. In particular, we focus on the importance of the minimum pacing timer, and show that a 10-millisecond timer would be too coarse for removing short-scale traffic burstiness, while a 1-millisecond timer would be sufficient to make the traffic almost as smooth as a Poisson stream in sub-RTT scales.

Categories and Subject Descriptors

C.2.m [Computer-Communication Networks]

General Terms

Measurement, Experimentation, Performance

Keywords

Traffic modeling, burstiness, wavelet-based multiresolution analysis, ON-OFF model, TCP self-clocking, TCP pacing.

*This work was supported by the NSF CAREER award ANIR-0347374.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

SIGMETRICS'05, June 6–10, 2005, Banff, Alberta, Canada.
Copyright 2005 ACM 1-59593-022-1/05/0006 ...\$5.00.

1. 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 [21] 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). The significance of these statistical properties for queueing performance and traffic prediction has been debated, with some models arguing for a major impact (see [6]), while others insisting that finite buffers limit the largest correlation scale¹ that can affect queueing (see [10]). 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 [31], also related to the distribution of transfer sizes and user thinking times [4].

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 [11, 9]. 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 in §2. 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.

Our objective in this paper is to examine the link between the TCP protocol and the short scale burstiness of Internet

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

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 queuing due to the packets of the flow itself or due to cross traffic, can shape the packet interarrivals of a TCP flow in an one-level or two-level ON-OFF pattern, respectively. This packet-train structure creates strong correlations and burstiness in 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 the commonly used 10-millisecond timer would be too coarse for removing short-scale traffic burstiness. On the other hand, a 1-millisecond pacing timer can reduce the burstiness in sub-RTT scales to almost the level of a Poisson process with the same average rate.

Note that we do not claim that TCP self-clocking is the only mechanism that can create short scale burstiness. Our results can help, however, to explain the results of previous measurement studies in terms of the characteristics (such as size, RTT, capacity, average window size) of the TCP flows that were dominant in the corresponding packet traces.

Section 2 summarizes the most relevant previous work. Section 3 gives a brief background on multiresolution analysis. Section 4 shows the link between TCP self-clocking and sub-RTT burstiness for a single TCP connection. Section 5 focuses on the effects of traffic aggregation. Section 6 is a case-study, in which we analyze an OC-48 trace and identify the flows that cause sub-RTT burstiness. Section 7 examines the effect of pacing. We conclude in Section 8.

2. PREVIOUS WORK

The work by Feldmann et al. [8] was one of the first to focus on short time scales, and also to use wavelet-based multiresolution analysis in the analysis of the *scaling behavior* (a special type of correlation structure) of Internet traffic. That work provided empirical evidence that WAN traffic can be modeled using a *multifractal model*, similar to that developed in [26]. More recent work, however, argues that the traffic at a tier-1 ISP is well modeled as monofractal, rather than multifractal [34]. 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 highly aggregated WAN links, the process can only exhibit monofractal scaling.

A paper that is more related to our work is [7]. The authors 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. They identified, however, *ACK compression* [33], 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 new insight in the observations of [7]. For instance, it now becomes clear that ACK compression is not necessary to create sub-RTT burstiness; self-clocking by itself is a sufficient condition, as long as the bandwidth-delay product is larger than the TCP window.

More recently, it has been shown that the short scale burstiness does not depend on the TCP flow arrival process [12]. Additionally, in not heavily loaded networks, correlations across different flows do not affect the short scale burstiness either. In a follow-up work [13], the authors showed that 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. Another related study was published in [34]. The authors 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 self-clocking, putting the results of [34] in a more TCP-specific context and providing a cause-effect relationship.

Liu and Baras proposed a structural model for TCP traffic, referred to as Hierarchical On-Off (HOO) model [24]. Misra and Gong had shown earlier that a hierarchical On-Off traffic pattern can produce scaling in a certain range of time scales [22]. HOO captures the transmission of packet bursts within a TCP window, and so it is similar with our model in §4. However, the analysis of [24] does not relate the transmission rate and length of those bursts with TCP’s self-clocking. Additionally, it does not explain how the relation between the window size and the flow’s bandwidth-delay product controls the TCP burstiness in sub-RTT scales.

Sarvotham et al. proposed an interesting classification of traffic in *alpha* and *beta* flows (analyzing the traffic at the 500ms time scale) [27]. The former are large transfers over high-capacity links and produce non-Gaussian traffic, while the latter are mostly low-throughput transfers and they produce Gaussian and LRD traffic. Jiang et al. 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 [18]. Those reasons include UDP message segmentation, TCP slow start, lost ACKs, and others.

Morris and Lin 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 [23]. Such a linear relation is characteristic of a Poisson process, but only for the given scale. In an extension of that work, it has been shown that the variance-mean relation depends on the network load and on the time scales of interest, and that a general characteri-

zation of the traffic as ‘‘Poisson-like’’ would not be accurate [32]. Recently, it has been argued that network traffic today can be well represented by the Poisson model in sub-second scales [29]. 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 [3].

3. 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. Furthermore, we define more precisely what we mean by ‘‘burstiness’’ and smooth versus bursty traffic. This tool is based on *wavelet-based MultiResolution Analysis (MRA)*, and it was developed by Abry and Veitch [1, 30]. 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 $T_j=2^jT_0$ ($j = 0, 1, \dots$) is a time series $X_j=\{X_{j,0}, X_{j,1}, \dots\}$, 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 [30], 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*². The Haar wavelet coefficients $W_{j,k}$ at a scale j are defined as

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

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

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

$$= 2^{-j}\text{Var}[\Delta X_{j-1,k}] \quad (3)$$

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 $\Delta X_{j-1,k}$

²Even though we explain energy plots in terms of the Haar wavelet for simplicity, we actually use higher order Daubechies wavelets.

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 [1].

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

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

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

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

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

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

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 [1]. 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*³. 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 actually bursty. 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. Our 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 four renewal

³A subtle point here is that we can only consider traffic processes with a well-defined rate. For instance, we cannot do these energy comparisons with renewal processes that have infinite variance interarrivals.

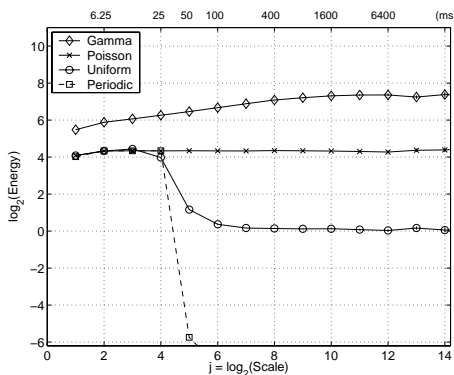


Figure 1: Energy plots for Exponential, Gamma, Periodic, and Uniform independent interarrivals.

processes (i.e., independent and identically distributed interarrivals). The corresponding interarrival distributions are Exponential, Gamma, Uniform, and Periodic. All four distributions have the same average interarrival (50ms), while the reference scale is $T_0=25$ ms. Because the average rate is the same in all four 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 [3]. Then, the variance $\text{Var}[\Delta X_{j-1,k}]$ 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 $L = 30$ ms and $U = 70$ ms. 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 ϵ 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 1 with $T=50$ ms. The periodic model can be viewed as the smoothest traffic process, because it has the minimum possible energy after scale T .

A traffic process that we use extensively in the rest of the paper is the *packet train* model [16]. In that model, an *ON-period* consists of W packets that arrive with a constant interarrival τ_o , causing a burst of duration $W\tau_o$. After every burst, an *OFF-period* follows with mean duration τ_f in which no packets are received. The average rate of this process is $W/(W\tau_o + \tau_f)$ packets per second. Figure 2 shows the energy plot for two packet train models in which the OFF-period is exponentially distributed. The train characteristics are $(W, W\tau_o, \tau_f)$ and they are set to $(8, 100$ ms, 300 ms) and

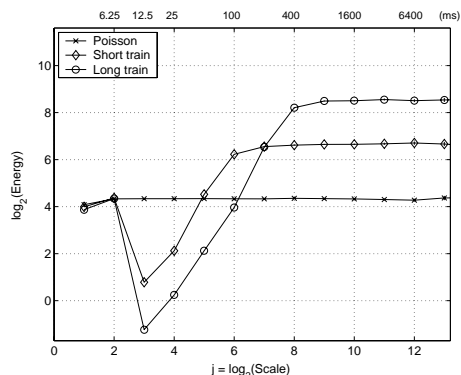


Figure 2: Energy plots for two packet train models.

(32, 400ms, 1200ms) for each train model, respectively. Note that both trains have the same packet transmission time $\tau_o=12.5$ ms, and the same average rate (1 packet per 50ms), but the second model has four times longer ON-periods. The following facts for the packet train model with exponential OFF-periods can be derived from Equation (2). First, the packet train model has a periodicity at τ_o (12.5ms), which becomes stronger as W increases, causing an energy drop at the corresponding scale. Second, the energy increases from τ_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. Third, 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. In the next section, we will see that TCP self-clocking can create, under a certain condition, a structure of packet interarrivals that resembles that of the packet train model in sub-RTT time scales.

4. 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 [15]. 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 ACK⁴. 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 explain that because of self-clocking a TCP sender can be trapped into a state of sending long

⁴There are several deviations from this behavior, including congestion window increases, idle times when the application has no data to send, and retransmission timeouts.

packet trains at the full capacity of the forward path. We do so in two parts: first, ignoring any cross traffic in the forward path, and second considering such traffic. Examples of TCP flows from traffic traces provide further insight in the structure of TCP packet interarrivals. Then, to further illustrate the link between self-clocking and burstiness, we create two simple and stationary models of TCP interarrivals in sub-RTT scales, and show, with MRA energy plots, that they create bursty traffic. For simplicity, the following models are open-loop, meaning that they do not consider the impact of the network state on the resulting TCP interarrivals, and so they should be considered only as a first-order approximation to self-clocking. Furthermore, we make several simplifying assumptions: ACKs are not affected by queuing 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.

4.1 Self-clocking without cross traffic

Consider a TCP flow with a minimum RTT T seconds. The capacity of the bottleneck link 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 can be larger than T when a queue builds up at 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 an interarrival $\tau=L/C$ between any two successive packets. The receiver will respond to each packet with an ACK, sent with the same interarrival τ . 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 interarrivals τ between successive packets, i.e., at the full capacity C of the forward path. With Delayed-ACKs, an ACK is generated in every 2τ seconds, but the interarrivals between successive data packets of the same round-trip is still τ because an ACK releases two back-to-back packets in that case.

Periodic interarrivals. 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 τ 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 $\tau=L/C$, as shown in Figure 3-a, and the energy plot becomes as in Figure 1. 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. Figure 4-a shows the distribution of packet interarrivals for a TCP flow (from the trace of Section 6) that falls in this class. The RTT for this flow is about 52ms, the packet transmission time is $L/C=1.2$ ms, and the average window size is about

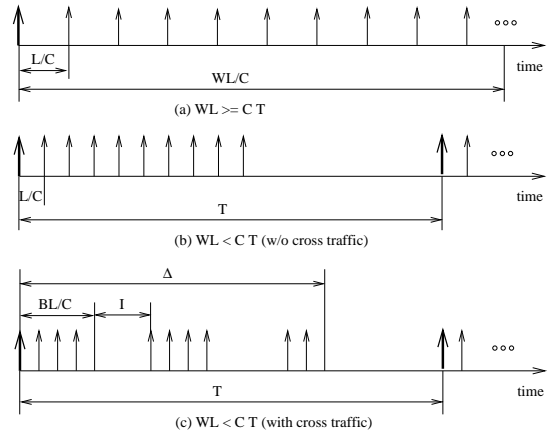


Figure 3: Packet interarrivals for a TCP flow during a particular round-trip of length T . Three cases are shown depending on whether $WL < CT$, and on whether there is cross traffic in the flow's bottleneck.

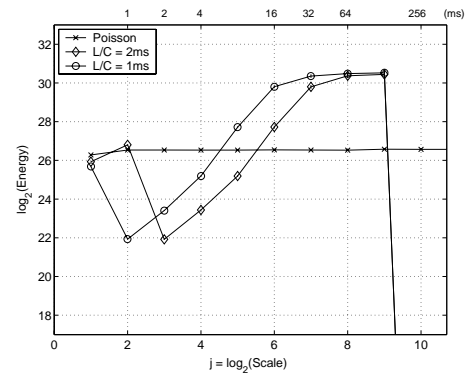


Figure 5: Energy plot for a periodic TCP self-clocking model without cross traffic.

42 packets. Note that almost all interarrivals are equal to L/C , which means that the flow is basically periodic.

One-level ON-OFF interarrivals. On the other hand, it is easy to see that if $W\tau < T$, the traffic process follows the packet train model of §3. Specifically, TCP generates during each round-trip k a packet train with size W_k packets, interarrivals of length $\tau = L/C$ during the train, and with an idle time $\tau_f = T - W_k\tau$ between trains. The timing-pattern of this traffic process is shown in Figure 3-b. Notice that TCP does *not* distribute its window throughout the RTT. Instead, when 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 structure* of packet interarrivals. Figure 4-b shows a TCP flow that falls in this class. The flow has $T=49$ ms, $L/C=1.2$ ms, $W=12$ packets (average). Note that about 90% of the interarrivals are spaced at about L/C , corresponding to interarrivals within each train. The remaining 10% of the values are spaced at about 36ms, which is roughly equal to $T - WL/C$, corresponding to interarrivals between trains in successive round-trips.

Figure 5 shows the MRA energy plot for a synthetically generated flow that follows the one-level ON-OFF model of

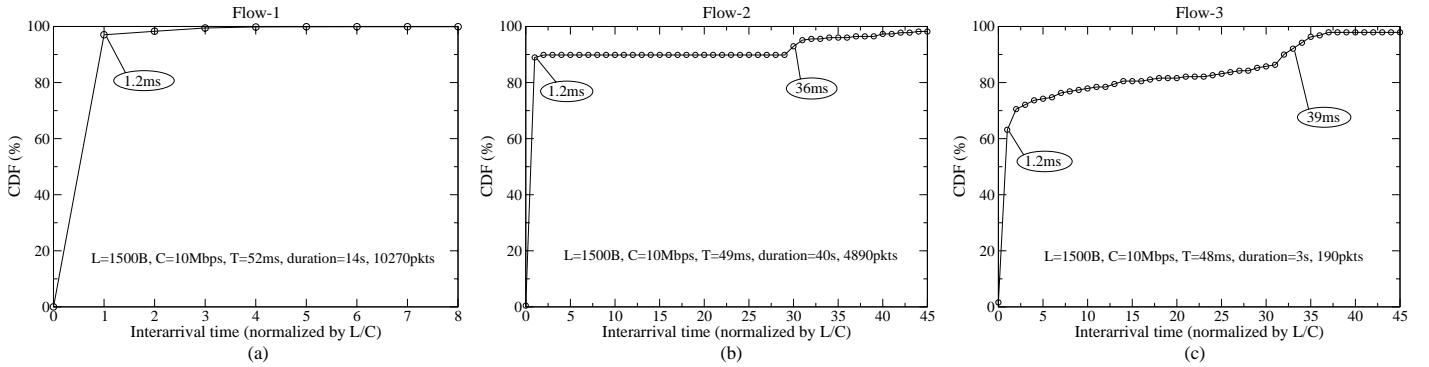


Figure 4: Distributions of TCP packet interarrivals in three TCP flows from the trace of Section 6. All three flows have the same packet transmission time $L/C=1.2\text{ms}$, corresponding to an Ethernet MTU packet ($L=1500\text{B}$) and a $C=10\text{Mbps}$ capacity. The interarrivals are rounded to the nearest multiple of L/C .

Figure 3-b. In this flow, we set $T=256\text{ms}$, $W=16$ packets, $L=1250$ bytes, and consider two values of τ : 1ms ($C=10\text{Mbps}$), and 2ms ($C=5\text{Mbps}$). 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 from 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 means that *if the capacity of a path increases, but with a constant TCP average throughput, the source creates bursty traffic over a wider range of sub-RTT scales.* This has a practical implication: as the capacity of the network edges increases, for instance replacing dial-up modems with DSL or cable modems, or replacing FastEthernet with Gigabit Ethernet, we should expect to see wider-range burstiness in short timescales if the maximum TCP window (often determined by the default socket buffer size at the end-hosts) remains the same. Finally, the energy drop at scale 10 corresponds to the RTT T and is due to the periodicity of the model at that scale.

4.2 Self-clocking with cross traffic

In the general case, the bottleneck \mathcal{B} may carry some cross traffic together with the TCP flow that we analyze. We assume that that TCP flow 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 $a_{i+1} - a_i = L/C_s$ ($i=1 \dots N-1$). 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 interarrival L/C . Otherwise, if one or more cross traffic packets arrive in that interval, the packets P_i and P_{i+1} will depart with a smaller rate than C , i.e., a larger interarrival than L/C . Such cross traffic interference partitions the TCP window during that round-trip to a number of smaller bursts. The length of these bursts, as well as the interarrivals between them, depend on the cross traffic load. The higher the load at the bottleneck, the shorter and more widely dispersed are the resulting TCP bursts. Fur-

thermore, the presence of bursts in a round-trip k can create a similar pattern in the corresponding ACKs, propagating the burstiness of round-trip k to the following round-trips.

In general, suppose that a TCP window of W packets during a particular round-trip is segmented 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 3-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 Δ . The expected value of Δ is a function of the average cross traffic rate, and it can be derived in certain cases as shown in [5]. Note that this structure is a *two-level ON-OFF pattern*. At the lower level, each ON duration consists of 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 Δ ($WL/C \leq \Delta < T$). Figure 4-c shows a TCP flow that falls in the two-level ON-OFF class. The flow has $T=48\text{ms}$, $L/C=1.2\text{ms}$, $W=6$ packets (average), and $\Delta=10\text{ms}$ (average). Note that about 65% of the interarrivals are spaced at about 1.2ms (L/C), corresponding to interarrivals within each burst. Another “mode”, with about 15% of the interarrivals, appears at about 39ms that is roughly equal to $T - \Delta$, corresponding to interarrivals between successive round-trips. The remaining values are distributed almost uniformly between 1.2ms and 37ms , corresponding to interarrivals between successive bursts of the same round-trip.

Figure 6 shows the MRA energy plot for a synthetically generated flow that follows the two-level ON-OFF model of Figure 3-c. In this flow, we set $W=16$ packets, $T=1228\text{ms}$, $L=1500$ bytes, $C=10\text{Mbps}$, $\Delta=128\text{ms}$, and $\bar{B}=2$ packets. The first five parameters are constant, while B_i follows a geometric distribution with mean \bar{B} ⁵. The burst interarrivals I_i are first generated by an exponential distribution with mean $\bar{I} = (\Delta - WL/C)/(K-1)$, and then scaled to meet the constraint $\sum_{i=1}^{K-1} I_i = \Delta - \frac{WL}{C}$, where K is the number of generated bursts in that round-trip. The energy plot of a Poisson process with the same average rate

⁵The last burst length B_K during a round-trip follows a truncated geometric distribution with maximum length W if $K=1$, and $W - (B_1 + \dots + B_{K-1})$ if $K > 1$. This constraint guarantees that the window will be W packets.

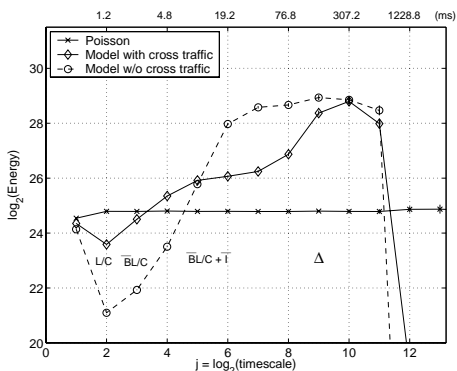


Figure 6: Energy plot for a periodic TCP self-clocking model with and without cross traffic.

WL/T is also shown for comparison. Note that the energy increases in two regions, corresponding to the two ON-OFF patterns: first, between L/C and BL/C , and then, between $BL/C + \bar{I}$ and Δ . 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 more bursty at the higher end of sub-RTT scales because its packet trains are not segmented into shorter bursts.

5. FLOW 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. In practice, however, most network links carry aggregates of some TCP and non-TCP flows. How does the burstiness of the aggregated flow relate to the burstiness of its constituents? And, does the “intermingling” of packets from different flows mitigate the correlations between packets of individual flows?

To answer the first question, note that the energy \mathcal{E} of the aggregation, or superposition, of several independent flows is equal to the sum of the energies of the constituent flows. Specifically, suppose that X and Y are two stationary, independent traffic processes, and that $Z = X + Y$ is the aggregate traffic process that results when X and Y are multiplexed on the same link. As long as the multiplexing process does not introduce queuing delays between the packets of the two processes, X and Y remain independent. With this assumption, it is easy to show from Equation (2) that $\mathcal{E}_j^Z = \mathcal{E}_j^X + \mathcal{E}_j^Y$ at any scale j . An implication of this result is that flows that do not have significant energy relative to the aggregate flow do not have a major impact on the burstiness of the latter.

To answer the second question, let us first review 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 [3]. It is important to note, however, that the previous result assumes that the rate of each constituent flow is λ/N , so that the rate of the aggregate flow remains the same, equal to λ , 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

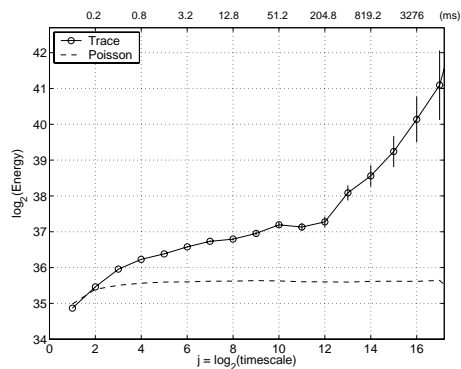


Figure 7: Energy plot of the MFN OC-48 trace.

average rate λ 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 [28]. 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 lose their correlation structure. The latter 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

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

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

Recently, Karagiannis et al. argued that Internet traffic today can be well-modeled by the Poisson model in sub-second scales [29]. The authors justified this position based on statistical analysis of the interarrivals in some OC-48 traces from MFN. Specifically, they observed that the interarrivals of the aggregate traffic match closely the exponential distribution, and also, the interarrivals appear to be independent. The approximate match to the Poisson model was attributed to the previously mentioned superposition result from the theory of point processes. However, there was no discussion regarding the deviations from the Poisson model shown in both the IDI graph (Index of Dispersion for Intervals) and the MRA energy plot. We analyzed one of the MFN OC-48 traces that were used in [29]; specifically, the trace collected between 11:00-11:10am on 1/15/2003. The trace consists of 96% TCP traffic, and has an average rate of 278 Mbps. The energy plot of the trace is shown in Figure 7, together with the energy of the corresponding Poisson process. The weighted average of the TCP flow RTTs (defined in §6) in this trace is about 172ms, meaning that the sub-

RTT scales extend up to scale-12 (200ms). We observe that the traffic of this trace is clearly more bursty than the Poisson model in sub-RTT scales, which contradicts the main result of [29].

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, as long as the aggregation time scale is sufficiently large (horizontal aggregation) and the number of flows is sufficiently large (vertical aggregation) [20]. Since both the mean and the variance of the marginal distribution increase proportionally to N , the coefficient of variation decreases with the square-root of N . So, the traffic appears to be “smoother” as N increases simply because the magnitude of the traffic variations at any given time scale decrease in magnitude relative to the average traffic rate. 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.

6. CASE STUDY

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_{irg}	2.22	3207	50.9	25.3
S_{sml}	0.03	7277	0.6	1.7
$S_{ir\ominus}$	2.12	3123	48.6	24.2
$S_{sm\ominus}$	0.10	84	2.3	1.1

Table 1: Subsets of the IPLS 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. The unidirectional trace that we use in this paper was collected, with a high-precision DAG 4.23 OC-48 monitor, at the OC-48 link that connects the Abilene routers from Cleveland (CLEV) to Indianapolis (IPLS). The trace was collected on 8/14/2002 between 9:00-11:00am, and it is publicly available at NLANR-MOAT [25]. The following results refer to a 90-second segment of the trace (10:08:30-10:10:00). In the energy plots of this section, the reference time scale T_0 is 0.1ms.

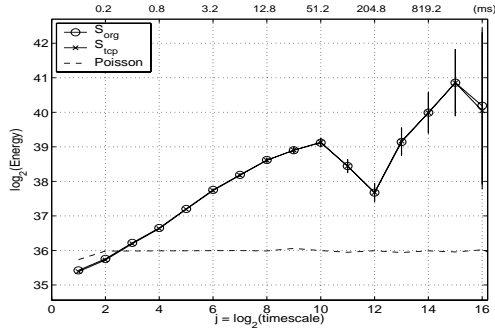
We start from the original trace, say S_{org} , and gradually form an increasingly narrower subset of flows that are responsible for the short scale burstiness of the entire trace. This investigative work, which is similar to gradually reducing a set of crime suspects through additional evidence, leads 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 §4 and §5, we find that this set includes *bulk TCP flows that*

have a large bandwidth-delay product relative to their window size. Table 1 shows the notation, and a few statistics, for the various sets of flows that we consider in the remaining of the section.

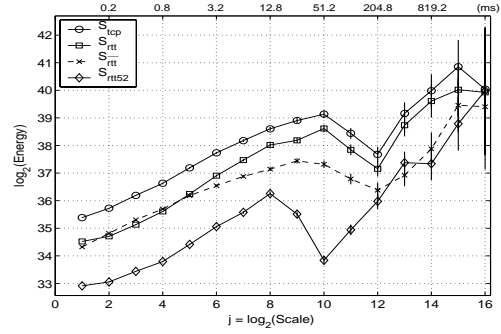
First, Figure 8-a shows the energy plot of the original trace S_{org} , together with the energy plot of a Poisson process with the same average rate. Note that the trace shows strong burstiness in short time scales, up to about 200ms. In longer scales, the trace exhibits a range of linearly increasing energy due to LRD effects, which is typical of Internet traffic (see [8] or [34] 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 8-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 [17] 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 8-b shows the energy plots of the two subsets, which are quite similar in shape and magnitude.

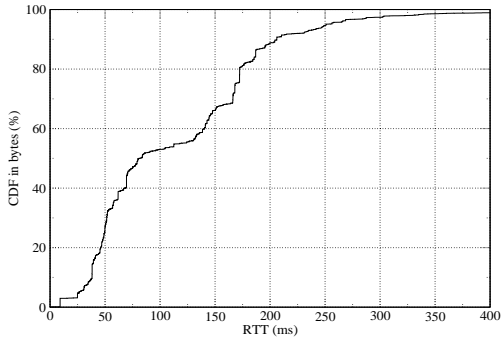
The estimated RTT distribution for S_{rtt} is shown in Figure 8-c, plotted in terms of bytes rather than flows. Note that most of the traffic in that subset is carried by flows with RTT in the range 25-250ms. The weighted average of the flow RTTs, with each RTT measurement weighted by the fraction of bytes in the corresponding flow, is about 117ms. Notice that the dip in the energy plot of Figure 8-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 §3, 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 8-b. Note that the energy plot dip has now moved to scale 10 (51ms), 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*



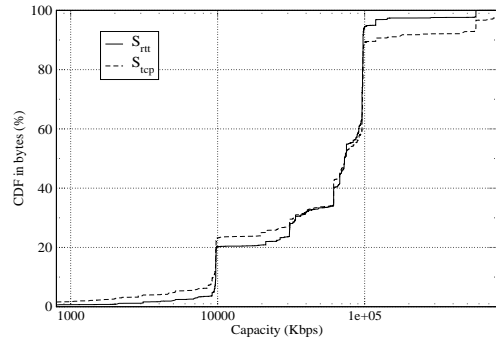
(a) Energy plots for Sorg and Step



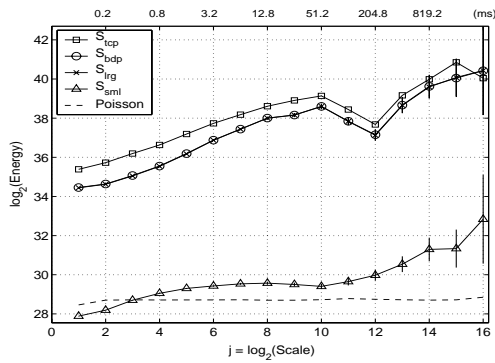
(b) Energy plots for Srtt, Srtt52, and Step



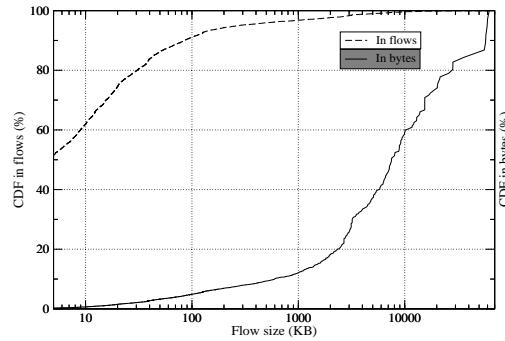
(c) RTT distribution for Srtt



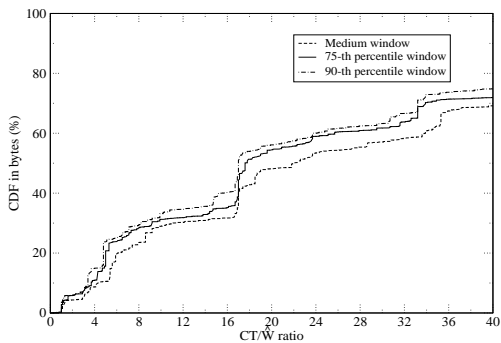
(d) Capacity distribution for Srtt and Step



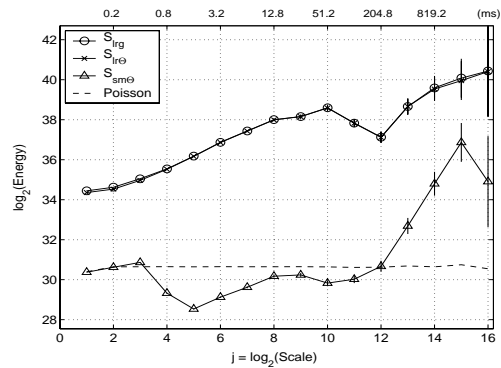
(e) Energy plots for Sbdp, Slrg, and Ssm1



(f) Flow size distribution for Sbdp



(g) Burstiness ratio distribution for Slrg



(h) Energy plots for Slrø and Ssmø

Figure 8: Analysis of Abilene OC-48 trace.

burstiness.

To estimate the bandwidth-delay product of a TCP flow, we need to know its forward-path capacity C , as defined in §4, 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-ACKs. Using a passive capacity estimation technique, similar to that reported in [19], we can estimate the capacity of about 90% of the bytes of S_{tcp} , and a similar fraction of S_{rtt} . Figure 8-d shows the capacity distribution for the two sets. Notice that about 40% of the bytes belong to high-capacity flows ($C \approx 100$ Mbps), while about 90% of the bytes are generated from flows with $C \geq 10$ Mbps. 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 8-e.

We next focus on the link between flow size and sub-RTT burstiness. The conjecture is that it is the large flows that determine the shape of the aggregate energy plot. Figure 8-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 the remaining small flows. Figure 8-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 beyond 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 *burstiness ratio* $\Theta = CT/\hat{W}$. This important metric is the ratio of the bandwidth-delay product of a TCP data flow to a “typical” window size \hat{W} for that flow. Figure 8-g shows the distribution of Θ 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 Θ . In the following, we define

\hat{W} based on the 75-th percentile.

An important observation in Figure 8-g is that Θ 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 $S_{lrg\Theta}$ and $S_{sml\Theta}$, where the former includes the TCP flows with $\Theta > 2.0$, while the rest of S_{lrg} forms $S_{sml\Theta}$. The energy plots of the two new subsets are shown in Figure 8-h. First, note that the energy plot of $S_{lrg\Theta}$ completely overlaps with S_{lrg} , verifying our main conjecture that *flows with large values of Θ determine the sub-RTT burstiness of the entire trace*. Second, perhaps more surprisingly, the energy plot of $S_{sml\Theta}$ is below the energy of a Poisson process with the same rate as that subset, i.e., *TCP flows with small values of Θ , 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 a burstiness ratio of at least two, i.e., more than 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 a burstiness ratio that is roughly equal to one produce Poisson-like, or even smoother, traffic.

We have repeated the previous analysis in several NLANR traces, ranging from OC-3 to OC-48 links. Consistently, and without a single exception, we found that all traces show sub-RTT burstiness, and that the burstiness is created by relatively large TCP flows (> 10 -20KB) with large burstiness ratio Θ . Actually, several traces did not include large flows with small burstiness ratio. We note, however, that the NLANR traces are collected mostly from university access links and research networks, and so they may not be representative of commercial Internet traffic in both the flow size distribution and the flow capacity distribution.

7. TCP PACING

The previous sections showed that self-clocking is largely responsible for the burstiness of individual TCP flows, and of 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. One way to remove, or at least reduce, the burstiness of a TCP flow is to perform *pacing* at the sender [2, 14]. 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,⁶ 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

⁶For simplicity we assume here that the window is measured in packets instead of bytes.

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 ( $n, X$ );
}

```

Figure 9: Pseudo-code for calculating X and n .

W and small T . For example, if $T=15\text{ms}$ and $W=100$ packets, the sending host would have to generate a timeout, to schedule a packet transmission, in every $150\mu\text{s}$. 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 9. X_{min} is the minimum burst size (for instance, 1-2 packets) imposed. 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.

We now examine the burstiness reduction with ideal pacing as well as with practical pacing for $T_c=10\text{ms}$ or 1ms and $X_{min}=2$ packets. We show the impact of pacing on the aggregate traffic, S_{bdp} from the OC-48 trace of §6, after performing pacing on each of the constituent flows. The RTT T and the sequence of window lengths $\{W_i\}$ for each round-trip i are estimated as explained in §6. The approach of artificially modifying the interarrivals of a packet trace is referred in the literature as *semi-experimental* [13]. We should note that a limitation of the semi-experimental approach is that it is intrinsically “open-loop”; specifically, it does not consider the impact of pacing on the dynamics of TCP flows (e.g., throughput, losses, RTT variations).

Figure 10 shows the resulting energy plots. The original trace (upper graph) has strong sub-RTT burstiness (scales 2-10). With ideal pacing, the traffic becomes smooth throughout the sub-RTT scales up to scale 10 (50ms), as we would expect, due to the periodic nature of packet departures in every round-trip. The two curves that correspond to practical pacing (lower graph) 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 aggregate up to scale 6, i.e., there is remaining burstiness in very small time scales (up to 3-4ms). The reason is that, with $T_c=10\text{ms}$, the burst length X can be large when the ratio T/W is less than T_c . On the other hand, the 1ms pacing timer manages to reduce burstiness in sub-RTT scales to almost the level of a Poisson process with the same average rate, even though it still does not generate smooth traffic, the way ideal pacing does. We note that pacing does not affect the burstiness of the traffic in time scales that extend beyond the RTT range (scale 11 and higher).

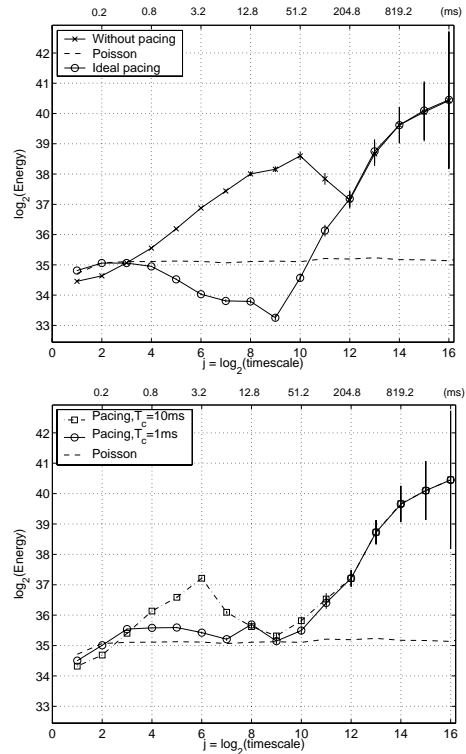


Figure 10: The effect of ideal and practical pacing on the S_{bdp} set of §6.

8. CONCLUSIONS

We showed that TCP self-clocking, coupled with queuing 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 (i.e., high burstiness ratio). 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 burstiness ratio of more than 2.0. We also showed that 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.

Acknowledgments

We are grateful to the NLANR MOAT traffic collection project⁷, to CAIDA for providing us with the MFN trace, and to Darryl Veitch for making his MRA analysis tool publicly available.

⁷Supported by the NSF cooperative agreements ANI-0129677 and ANI-9807479.

9. REFERENCES

- [1] P. Abry and D. Veitch. Wavelet Analysis of Long-Range Dependent Traffic. *IEEE Transactions on Information Theory*, 44(1):2–15, Jan. 1998.
- [2] A. Amit, S. Savage, and T. Anderson. Understanding the Performance of TCP Pacing. In *Proceedings of IEEE INFOCOM*, Apr. 2000.
- [3] D. R. Cox and V. Isham. *Point Processes*. Chapman and Hall, London, 1980.
- [4] M. E. Crovella and A. Bestavros. Self-Similarity in World Wide Web Traffic: Evidence and Possible Causes. *IEEE/ACM Transactions on Networking*, 5(6):835–846, Dec. 1999.
- [5] C. Dovrolis, P. Ramanathan, and D. Moore. Packet Dispersion Techniques and a Capacity Estimation Methodology. *IEEE/ACM Transactions on Networking*, 12(6):963–977, Dec. 2004.
- [6] A. Erramilli, O. Narayan, and W. Willinger. Experimental Queueing Analysis with Long-Range Dependent Packet Traffic. *IEEE/ACM Transactions on Networking*, 4(2):209–223, Apr. 1996.
- [7] 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.
- [8] 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.
- [9] D. R. Figueiredo, B. Liu, and V. M. and D. Towsley. On the Autocorrelation Structure of TCP Traffic. *Computer Networks Journal, Special Issue on “Advances in Modeling and Engineering of Long-Range Dependent Traffic”*, 2002.
- [10] M. Grossglauser and J.-C. Bolot. On the Relevance of Long-Range Dependence in Network Traffic. *IEEE/ACM Transactions on Networking*, 7(5):629–640, 1999.
- [11] L. Guo, M. Crovella, and I. Matta. Corrections to “How Does TCP Generate Pseudo-Self-Similarity?”. *ACM SIGCOMM Computer Communication Review*, 32(2):30–30, 2002.
- [12] N. Hohn, D. Veitch, and P. Abry. Does Fractal Scaling at the IP Level Depend on TCP Flow Arrival Processes? In *Proceedings of Internet Measurement Workshop (IMW)*, Nov. 2002.
- [13] 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”*, 51(8):2229–2244, 2003.
- [14] D. R. J. Kulik, R. Coulter and C. Partridge. Paced TCP for High Delay-Bandwidth Networks. In *Proceedings of IEEE GLOBECOM*, Dec. 1999.
- [15] V. Jacobson. Congestion Avoidance and Control. In *Proceedings of ACM SIGCOMM*, Sept. 1988.
- [16] R. Jain and S. A. Routhier. Packet Trains - Measurements and a New Model for Computer Network Traffic. *IEEE Journal on Selected Areas in Communications*, 4(6):986–994, Sept. 1986.
- [17] H. Jiang and C. Dovrolis. Passive Estimation of TCP Round-Trip Times. *ACM Computer Communication Review (CCR)*, Aug. 2002.
- [18] H. Jiang and C. Dovrolis. Source-Level Packet Bursts: Causes and Effects. In *Proceedings of Internet Measurement Conference (IMC)*, Oct. 2003.
- [19] H. Jiang and C. Dovrolis. The Effect of Flow Capacities on the Burstiness of Aggregated Traffic. In *Proceedings of Passive and Active Measurements (PAM) workshop*, Apr. 2004.
- [20] J. Kilpi and I. Norros. Testing the Gaussian Approximation of Aggregate Traffic. In *Proceedings of Internet Measurement Workshop (IMW)*, Nov. 2002.
- [21] 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*, 2(1):1–15, Feb. 1994.
- [22] V. Misra and W. B. Gong. A Hierarchical Model for Teletraffic. In *Proceedings of 37th Annual IEEE Conference on Decision and Control*, 1998.
- [23] R. Morris and D. Lin. Variance of Aggregated Web Traffic. In *Proceedings of IEEE INFOCOM*, Apr. 2000.
- [24] N. X. Liu and J. S. Baras. Modeling Multi-Scale TCP Traffic With a Structural Model. *IEEE/ACM Transactions on Networking*, Apr 2003.
- [25] NLANR MOAT. Passive Measurement and Analysis. <http://pma.nlanr.net/PMA/>, Dec. 2003.
- [26] 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*, 45(3):992–1019, Apr. 1999.
- [27] R. R. S. Sarvotham and R. Baraniuk. Connection-level Analysis and Modeling of Network Traffic. In *Proceedings of Internet Measurement Workshop (IMW)*, Nov. 2001.
- [28] K. Sriram and W. Whitt. Characterizing Superposition Arrival Processes in Packet Multiplexers for Voice and Data. *IEEE Journal on Selected Areas in Communications*, 4(6):833–846, 1986.
- [29] M. F. T. Karagiannis, M. Molle and A. Broido. A Nonstationary Poisson View of Internet Traffic. In *Proceedings of IEEE INFOCOM*, Mar. 2004.
- [30] D. Veitch. Code for the Estimation of Scaling Exponents. <http://www.cubinlab.ee.mu.oz.au/~darryl>, July 2001.
- [31] 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*, Sept. 1995.
- [32] 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.
- [33] L. Zhang, S. Shenker, and D. D. Clark. Observations on the Dynamics of a Congestion Control Algorithm. In *Proceedings of ACM SIGCOMM*, Sept. 1991.
- [34] 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.