Spectral Characteristics of Saturated Links

Xinming He, Christos Papadopoulos, John Heidemann, Alefiya Hussain
Computer Science Department, University of Southern California
941 W.37th Place, Los Angeles, CA 90089
xhe@usc.edu, {christos,johnh,hussain}@isi.edu

***Abstract*— Internet protocols frequently create periodic patterns in traffic. Examples included packets paced by bottleneck links, periodic exchange of information such as routing, transport-layer effects such as TCP self-clocking, and application-level effects. Although measurement of such periodicities could shed light on Internet traffic, current understanding of periodic behavior in general traffic is quite limited. This paper explores this area by studying the spectral behavior of these kinds of traffic. Our technique is completely passive and can be applied to aggregate traces gathered at various observation points on the network. Unlike techniques measuring packet inter-arrival time, our technique does not require per-flow separation. Our experiments show that the signature of a saturated link persists in the presence of background traffic or when we observe only a portion of the traffic through the saturated link. We investigate how such signatures evolve as the traffic traverses through the network and identify the major influential factors that affect the signatures. Developing a technique to detect saturated links is part of our future work.**

***Index Terms*— Spectral Analysis, Network Traffic Analysis, Saturated Links**

I. Introduction

There exist several processes that govern the generation and shaping of Internet traffic. Some of these processes are periodic and operate at all communication layers: at the link layer, periodicities are imposed due to fixed link speeds; at the protocol layer, due to behavior such as windowing mechanisms and other periodic protocol operations such as routing updates; and at the application layer, due to behavior such as continuous media transmission. Such periodic processes imprint a *unique periodic signature* on their traffic. Periodicities are visible at several timescales, ranging from microseconds (e.g., clocking out packets on gigabit links) to days and years (e.g., diurnal cycles to seasonal traffic variations).

Studying such periodicities may provide useful information about the health of a network. For example, a highly utilized transit or peering link will impose a strong frequency proportional to the link speed and inversely proportional to the average packet size. This signal may be analyzed to distinguish a denial-of-service attack from congestion due to high normal traffic load. Typical attacks use very small packet sizes and thus an attack would impose a much higher frequency compared to normal traffic of similar intensity Another example is detecting attacks attempting to overload a web server through repeated requests. A machine carrying out this attack will exhibit a strong frequency in requests.

Unlike traditional network analysis techniques, spectral techniques focus on the periodic behavior of a phenomenon and are arguably more informative when analyzing dynamic behavior. Spectral analysis is a mature field used in statistics for several decades to detect hidden patterns and trends in time-series. Such techniques, however, have not been widely applied to the analysis of aggregate network traffic. Recent work presents strong evidence that applying such techniques to the analysis of network traffic is a very promising approach to study denial-of-service attacks [1], [2], DNS traffic behavior [3], traffic anomalies [4], and even protocol behavior in encrypted traffic [5]. Although this work has begun to explore the area, there has been relatively little work in applying spectral analysis to “typical” network conditions.

In this paper we use spectral analysis to study the signatures of saturated links. Our long-term goal is to develop a tool that can examine aggregate traffic to identify flows that pass through saturated links, even if the problem is several hops away and obscured by cross-traffic. Automating such a tool is future work; the immediate goal explored in this paper is to understand when a known signal is observable in the spectra of aggregate traffic when confronted by these challenges. Such an approach would be advantageous compared to current techniques such as SNMP data since it is based on passive measurement, and compared to inter-arrival studies [6] since it does not require separating traffic by flow and full spectral analysis can capture more information from the traffic arrival process.

In our approach, we first collect the time-stamped packet trace at an observation point, sample it based on an appropriate sampling rate to produce a time-series, and then use discrete Fourier transformation to retrieve prominent frequencies in the power spectrum which reflect periodic phenomena on the network traffic. At a high level, we explore the following questions: (a) are spectral techniques capable of capturing periodic phenomena on the network? (b) How do they compare to current techniques, such as histograms of packet inter-arrival time? (c) What are the influential factors on the power spectrum and what is their impact?

II. Methodology

A. Spectral Analysis

We use the following methodology to analyze the spectral characteristics of the traffic stream that traverses a saturate link, which is based on that proposed by Hussain et al. [1]. First, we use tcpdump to capture timestamped packet traces from the network. Then we divide each packet trace into slices of length L. The length of each slice is a configurable parameter and we will discuss shortly how to select it.

For each slice, we select a proper sampling rate p and define the packet arrival process $x(i)$ as the number of packets that arrive in the time period $[\frac{i}{p}, \frac{i+1}{p})$ where the time is relative to the start of the slice, and n varies from 0 to $L \times p$. This results in $N = L \times p$ number of samples in each slice. The selection of a proper sampling rate is another configurable parameter that we will discuss shortly. In addition, we subtract the mean arrival rate before proceeding with spectral analysis in the next step. The mean value results in a large DC component in the spectrum that does not provide useful information for our purposes.

After obtaining a time-series with N samples, we compute the power spectral density by performing the discrete-time Fourier transform on the autocorrelation function (ACF) of the packet steam. The autocorrelation is a measure of how similar the steam is to itself shifted in time by offset k [7], [8]. When $k = 0$ we compare the packet stream to itself, and the autocorrelation is maximum and equals to the variance of the packet stream. When $k > 0$ we compare the packet stream with a version of itself shifted by lag k. The autocorrelation sequence $r(k)$ at lag k is

$$c(k) = 1/N \sum_{t=0}^{N-k} (x(t) - \bar{x})(x(t+k) - \bar{x}); \qquad (1)$$

$$r(k) = c(k)/c(0) \qquad (2)$$

where $\bar{x}$ is the mean of $x(t)$ and N is the number of samples of the packet stream $x(t)$. The power spectrum (PSD) $S(f)$ of the packet stream is obtained by applying discrete-time Fourier transform to the autocorrelation sequence of length M:

$$S(f) = \sum_{k=0}^{M} r(k) e^{-\imath 2\pi f k} \qquad (3)$$

Meanwhile, we calculate the cumulative spectrum P(f) as the power in the range 0 to f, and normalize P(f) by the total power to get the normalized cumulative spectrum (NCS) C(f).

$$P(f) = \sum_{i=0}^{f-1} \frac{S(i) + S(i+1)}{2} \qquad (4)$$

$$C(f) = \frac{P(f)}{P(f_{max})} \qquad (5)$$

Intuitively, the spectrum $S(f)$ captures the power or strength of individual observable frequencies embedded in the time series, while the normalized cumulative spectrum $C(f)$ shows their relative strength. The spectrum can be compared both across time for consecutive slices gathered at the same point and across space for slices gathered at different points on the network to study how it evolves across time and across the network.

B. Parameter Selection

There are two important parameters in our technique, the length of each trace slice L, and the sampling rate p. If the slice length is too short, the spectrum will be sensitive to temporary or transient phenomena on the network. If it is too long, the arriving process is unlikely to be stationary. Since we target the spectral characteristics of a saturated link, we use a default value of 5 seconds for the slice length.

The sampling rate p is another important parameter. Given a sampling rate p, the highest frequency that is observable is $\frac{1}{2p}$ according to the Nyquist Theorem. If the sampling rate is too low, aliasing can occur. Too high a sampling rate incurs both storage and processing overhead.

For a given link speed and packet size, one can compute the maximum required sampling rate by computing the minimum packet inter-arrival time and sampling at twice that frequency. A more thorough exploration of varying sampling rate is the subject of future work. In this paper, unless otherwise stated we select a conservative rate of 100kHz, which is sufficient to observe 1500 byte packets over a 100Mb/s Ethernet.

III. Saturated Links in Simple Lab Scenarios

Our work is based on the assumption that a highly utilized link will impose a distinct signature on traffic traversing the link. In this section we validate our assumption by conducting experiments In simple scenarios to demonstrate the regularity imposed by saturated links. We first carry out the experiments with simple topologies and either zero or light cross traffic. We begin by examining a saturated link in a simple testbed to investigate the effects of simple and more complex traffic that saturates a link.

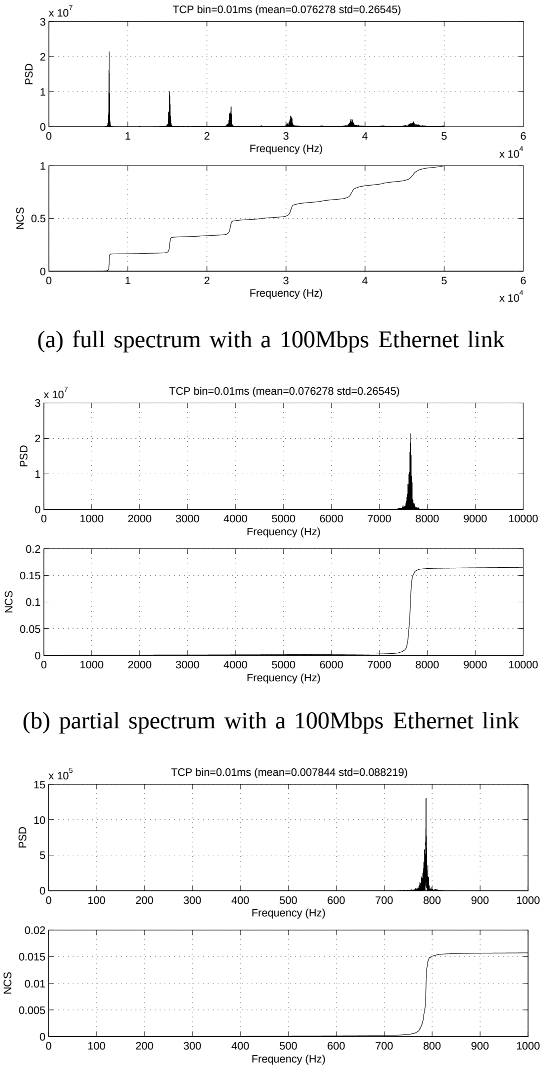
For these experiments we use two traffic generation tools, namely Surge [9] and Iperf [10]. Surge is used to generate synthetic web traffic and Iperf generates controlled TCP and UDP streams, which, for example, can mimic file downloads (TCP mode) or Constant Bit Rate traffic (UDP mode). We use tcpdump to record traffic at various observation points, such as the client, the server, or some intermediate point. A typical experiment lasts for 30 seconds and contains six 5-second long slices. Although there is some variation in the power spectra for the six slices, those are small.

A. Single Flow Traffic

We first investigate the spectrum when a single flow saturates the bottleneck. The experiment topology consists of two PCs connected to the same 100Mbps LAN and a single Iperf TCP flow running between them and ensuring that TCP is not window limited and captures nearly the entire link bandwidth. The observation point is attached to the the same LAN and passively listens to the all the traffic on the LAN.

Figure 1(a) shows the full spectrum we observe. This specific example depicts a single TCP flow using 1500 byte packets saturating a 100Mbps link. In addition to the fundamental frequency at 7630KHz, the spectra contain harmonics at multiples of this value. These harmonics consume a large portion of the energy in the spectrum. Since at this stage we are not proposing an algorithm to detect the presence of a saturated link but simply attempt to show that its signature persists, we first detect the fundamental frequency visually and then, for clarity, zoom our graphs to the interesting portion of the spectra, where the fundamental frequency of the saturated link lies, as shown in Figure 1(b). Developing an algorithm that detects a saturated link of unknown capacity is part of our ongoing work.

Figure 1(c) shows the resulting power spectrum for a 10Mbps link. The fundamental frequency 784Hz, which is close to the theoretical limit (833 Hz) imposed by a 10Mbps link with 1500-byte packets. We thus conclude that our methodology is capable of capturing the packet transmission frequency at a saturated link.



(a) full spectrum with a 100Mbps Ethernet link

(b) partial spectrum with a 100Mbps Ethernet link

(c) partial spectrum with a 10Mbps Ethernet link

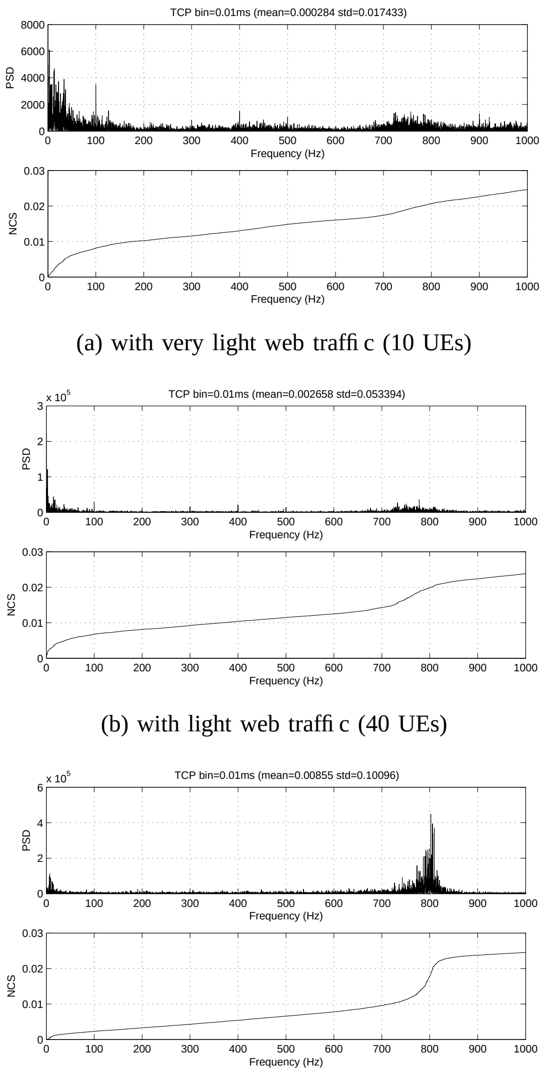
Fig. 1. Spectral signatures of different speed links

B. Multi-flow Traffic

It is rare for a single flow to saturate high-speed links. In this section experiments we replace the single TCP flow with web-like traffic. Web traffic imposes additional periodicities at the application layer and we are interested in examining the resulting signature. Since the link utilization depends on how many web flows are present, we conduct three experiments, with light, medium and heavy web traffic through the link.

We use the 10Mbps Ethernet topology and we generate web-like traffic using Surge [9]. In the first experiment we generate light web traffic by configuring surge to emulate 10 “user equivalents” (UEs). The resulting throughput is 0.15Mbps. Figure 2(a) shows the corresponding power spectrum, suggesting three observations. First, there is no appreciable signal strength at the dominant link frequency. Second, there are peaks at low frequencies (well under 100Hz), which correspond to the web requests and TCP window behavior due to Surge and TCP. Third, none of the peaks in the spectrum has particularly high absolute power. We suggest that this spectrum is typical of a lightly loaded link.

In Figure 2(b) and 2(c) we increase the load by a factor of 4 and 32 (to 40 and 320 UEs, respectively). The throughput of the web traffic is now 2.07Mbps and 8.21Mbps, respectively. First we observe that the absolute strength of spectral behavior is now two orders of magnitude higher than at light utilization. Second, we observe that as load rises, the dominant link frequency becomes stronger. At heaviest load there is a noticeable signal around 800Hz, similar to our single TCP flow. This experiment demonstrates that the dominant link signature is independent of number of flows. Instead it depends only on link bandwidth and packet size distribution. This is consistent with our first-principles reasoning that the dominant link frequency represents transmission of back-to-back packets.



(a) with very light web traffic (10 UEs)

(b) with light web traffic (40 UEs)

(c) with heavy web traffic (320 UEs)

Fig. 2. Spectral signatures at different traffic loads

IV. Saturated Links in Complex Scenarios

In the previous section we investigated the spectral characteristics of traffic on a saturated link when we could directly observe that traffic. Since congestion often occurs deeper in the network than the first hop, this section evaluates spectral behavior with different observation points and in the presence of competing traffic.

A. Classifying cross-traffic

Competing traffic will influence the power spectrum of a saturated link. In Figure 3 illustrates a network that illustrates several classes of cross traffic that might affect our observations. We assume traffic travels from source S to destination D passing through a bottleneck between R1 and R2, observed at node O. We are interested in observing the bottleneck signal generated at the R1–R2 link at the observation point. We identify three classes of cross-traffic:

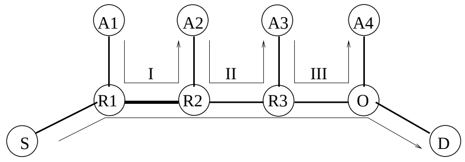


Fig. 3. Definitions of types of cross traffic

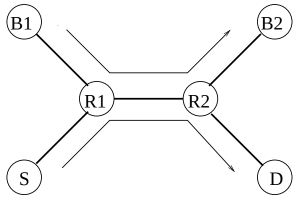


Fig. 4. Testbed topology

- *Type I (unobserved bottleneck traffic)* cross traffic that traverses the saturated link but does not reach our observation point. Such traffic carries part of the energy of the signature imposed by the bottleneck. Missing this traffic means possible attenuation of the signal at our observation point.
- *Type II (unobserved non-bottleneck traffic)* Cross traffic that is introduced after the saturated link, but is not observed at the observation point. Such traffic can distort the signal of the saturated link.
- *Type III (observed non-bottleneck traffic)* cross traffic that does not go through the saturated link but reaches our observation point. This traffic may introduce a false signature.

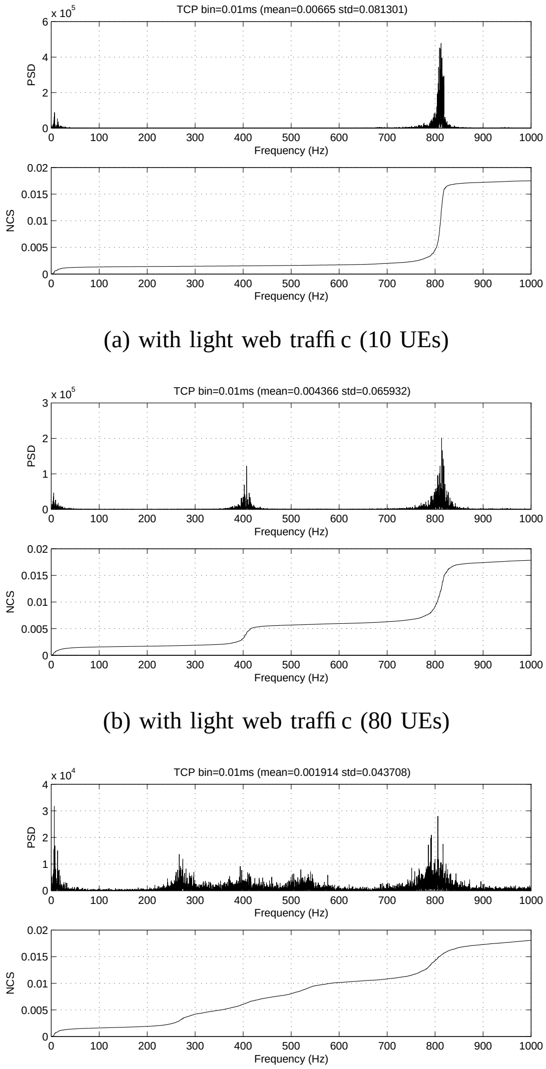
In the remaining of this section we carry out experiments to investigate the effect of these kinds of cross-traffic.

B. Testbed Experiments

To evaluate the impact of the three types of cross traffic identified above, we use the dumbbell topology shown in Figure 4.

For the first set of experiments, we investigate the impact of Type I cross traffic. We set the capacity of all links to 10Mbps. There are two types of traffic, a single Iperf TCP flow from node S to D and web traffic generated by surge between nodes B1 and B2. The observation point is at the link R2-D. We vary the number of web users in surge to control the volume of Type I traffic competing with the Iperf TCP flow on link R1-R2.

Figure 5 shows the power spectrum observed at D when the number UEs in Surge vary from 10 to 640. The corresponding throughput at link R1-R2 is always around 8Mbps, while the throughput at link R2-D is decreases from 8.1Mbps to 5.3Mbps and 2.3Mbps as cross-traffic increases We can see that as we increase the volume of Type I cross traffic, the energy around 800Hz becomes

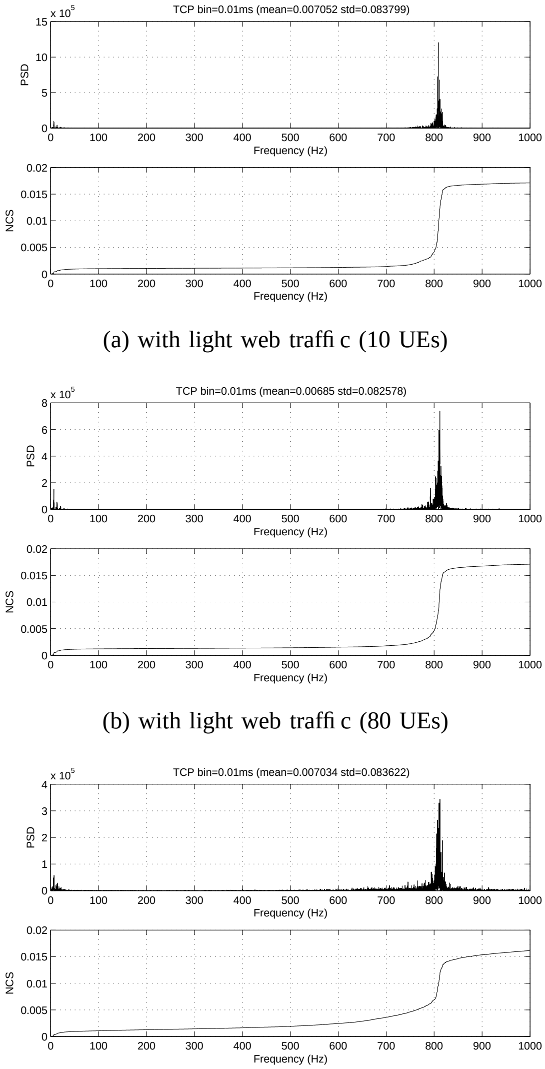


(a) with light web traffic (10 UEs)

(b) with light web traffic (80 UEs)

(c) with heavy web traffic (640 UEs)

Fig. 5. Power spectra as Type I cross-traffic increases



(a) with light web traffic (10 UEs)

(b) with light web traffic (80 UEs)

(c) with heavy web traffic (640 UEs)

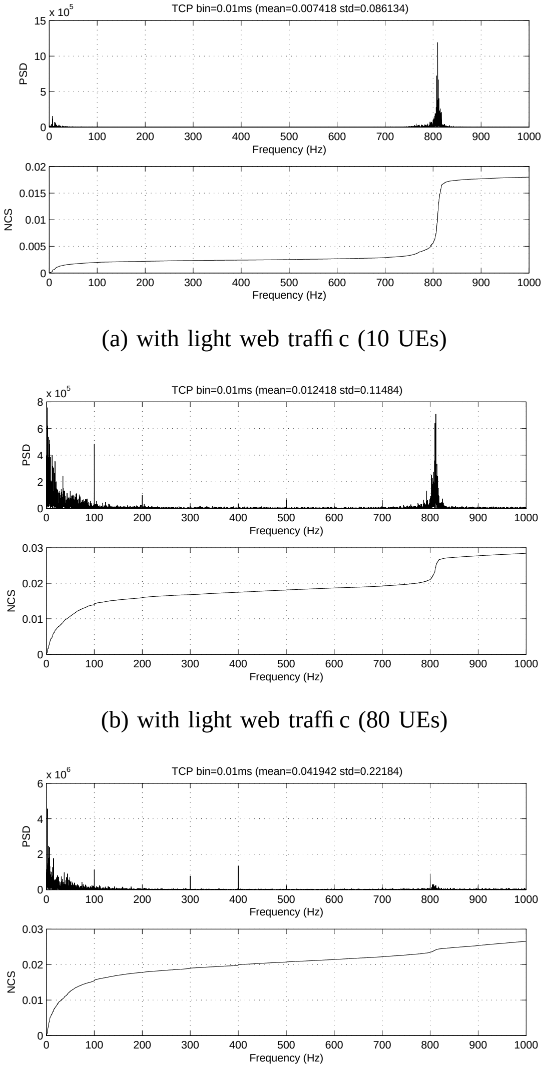
Fig. 6. Power spectra as Type II cross-traffic increases

weaker, but still visible. In addition, there is a new spike around 400Hz caused by packets experiencing queuing delay at the link R1-R2. The presence of energy at 400Hz is indicates of contention at the bottleneck link due to Type I cross-traffic. This phenomena was previously observed in studies of packet inter-arrival times [6], this corresponds frequency corresponds to the gap caused when one full-size packet queues behind another.

For the second set of experiments, we investigate the effect of Type II traffic. We set the capacity of the S-R1 link to 10Mbps, and the capacity of all other links to 100Mbps. We use the same two traffic sessions, with an Iperf TCP flow from S to D, and web traffic between B1 and B2. The observation point is again on the link R2-D and we vary the number of UEs to control the volume of Type II cross traffic.

Again Figure 6 shows the spectra at link R2-D as load rises. The corresponding throughput at link R1-R2 is 8.78Mbps, 13.41Mbps, and 38.9Mbps, respectively, and the throughput at link R2-D is steady around 8.3Mbps. We observe that as we increase the volume of Type II cross traffic the energy around 800Hz does not change much, although the spread of the signal grows slightly at higher loads. This result is because the R1-R2 link is not saturated and so only a few packets experience queuing delay there.

Finally we consider the effects of Type III traffic. We use exactly the same setting in the second set of experiments, but move the observation point to the link R1-R2. Figure 7 shows the results at R1-R2 as load grows. We observe the following. First, the energy over all the frequency spectrum has increased, as expected. Second, although the relative visibility of the 800Hz signal is greatly reduced (both in the PSD and the NCS), it is still observable in both and has relatively strong absolute power in the PSD.



(a) with light web traffic (10 UEs)

(b) with light web traffic (80 UEs)

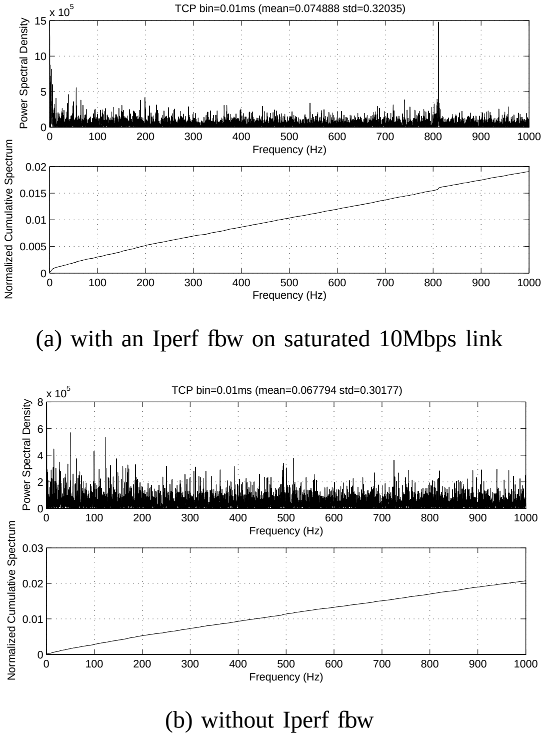
(c) with heavy web traffic (640 UEs)

Fig. 7. Power spectrum as Type III cross-traffic increases

C. *Wide-area network experiments*

Although the testbed provides a good environment for controlled experiments, the topology is still quite limited. We next validate our testbed observations on the Internet to consider a wide-area, multi-hop topology and richer, live background traffic. We placed our observation point on a router at the edge of USC, mirroring traffic from Internet-2. Our test flow was sent the University of Santa Barbara from a PC connected to a 10 Mbps LAN, over Internet-2, to a host at USC.

Figure 8(a) shows the power spectrum of the aggregate traffic observed at the monitoring link. The throughput of the TCP flow was 9.7Mbps, suggesting the bottleneck is the LAN at the source. The aggregate traffic at our monitoring host was 24.2Mbps. The PSD of Figure 8(a) shows a spike around 800Hz in the PSD suggests that the bottleneck traffic is visible, even *mixed with* aggregate traffic, although it is relatively small in NCS. For comparison Figure 8(b) shows aggregate traffic alone (taken just after our experiment), showing that the strongest individual peak (around 40Hz) is less than half the strength of our bottleneck flow.



(a) with an Iperf flow on saturated 10Mbps link

(b) without Iperf flow

Fig. 8. Power spectra of aggregate traffic at USC Internet-2 link

In this experiment the bottleneck flow was a relatively large part of aggregate traffic (about 40%). In future work we plan to investigate how visible the flow is when surrounded by greater traffic.

V. Related Work

Much prior work has studied network traffic to infer certain network properties, including packet delay and loss [11], link capacity and available bandwidth, bottleneck sharing among flows [6], network anomaly [4], and denial-of-service attacks [1]. In general, these measurements can be classified as either active measurements or passive measurements. Active measurements typically introduce probe traffic into the network for the study, while passive measurements utilize existing network traffic.

Examples of active measurements include pathchar, pathload, and Spruce, which are used to infer (available) bandwidth of a path. They all send out probes into the network and infer the link capacity or the available bandwidth of a path. Pathchar sends different size packets and utilizes the delay information to infer the hop-by-hop bandwidth. Pathload infers the available bandwidth along a path by adaptively saturating the path with packet trains. Spruce, on the other hand, takes representative samples of the state of the bottleneck queue by carefully choosing the initial spacing between probing packet pairs. Unlike these tools, our approach is passive.

Passive measurements typically gather traces of existing network traffic and analyze them using various methodologies. Katabi and Blake use packet inter-arrival times to infer the path characteristics such as bottleneck capacity, and bottleneck sharing among flows based the observation that the entropy of packet inter-arrival times is much lower for flows sharing the same bottleneck [6]. Unlike this approach, we compute the FFT of traffic. On one hand, this is more expensive prospect (although likely amenable to hardware assistance). The advantage is that spectral analysis can operate on aggregate traffic rather than per-flow traffic, as shown in Figure 7.

In recent years, a number of methodologies based on spectral techniques have been proposed to analyze network traffic. Hussain et al. apply spectral techniques to packet arrival time series to distinguish single-source and multi-source DDoS attacks [1], and more recently have extended this approach to attack re-identification [2]. Barford et al. use wavelets to analyze flow-level information to identify frequency characteristics of Dos attacks and other network anomalies [4]. Spectral analysis has been used to identify normal TCP traffic which exhibit strong periodicity around its round-trip time, whereas attack traffic does not [12]. Partridge et al. apply the Lomb periodogram technique to retrieve periodicities in wireless communication, including CBR traffic rate and the periodicity around FTP round-trip times [5]. We build on the methodology developed in this prior work, but apply spectral analysis to bottleneck link detection.

VI. Discussion and Future Work

Our preliminary experiments have shown that spectral analysis techniques are able to detect periodic phenomena in computer networks such as the frequency of a saturated link. Unlike techniques based on packet inter-arrival times, spectral techniques can be used to analyze aggregate traffic to uncover buried periodic phenomena. This makes these techniques a quite powerful tool in analyzing network traffic. As future work we plan to develop techniques to detect saturated links and investigate other periodic traffic phenomena.

References

[1] A. Hussain, J. Heidemann, and C. Papadopoulos, "A Framework for Classifying Denial of Service Attacks," in *Proceedings of the ACM SIGCOMM'2001*, Karlsruhe, Germany, August 2003.

[2] ——, "Identification of Repeated Attacks Using Network Traffic Forensics," under submission.

[3] A. Broido, E. Nemeth, and kc Claffy, "Spectroscopy of DNS Update Traffic," in *Proceedings of the ACM SIGMETRICS*, San Diego, CA, June 2003.

[4] P. Barford, J. Kline, D. Plonka, and A. Ron, "A Signal Analysis of Network Traffic Anomalies," in *Proceedings of the ACM SIGCOMM Internet Measurement Workshop*, Marseilles, France, November 2002.

[5] C. Partridge, D. Cousins, A. Jackson, R. Krishnan, T. Saxena, and W. T. Strayer, "Using Signal Proceesing to Analyze Wireless data Traffic," in *Proceedings of ACM workshop on Wireless Security*, Atlanta, GA, Sept. 2002, pp. 67–76.

[6] D. Katabi and C. Blake, "Inferring Congestion Sharing and Path Characteristics from Packet Interarrival Times," MIT, Tech. Rep. LCS Technical Report, 2001.

[7] G. Box, G. Jenkins, and G. Reinsel, *Time series analysis: forecasting and control*. Prentice-Hall, 1994.

[8] R. Bracewell, *The Fourier Transform and Its Applications*. McGraw-Hill, 1986.

[9] P. Barford and M. Crovella, "Generating Representative Web Workloads for Network and Server Performance Evaluation," in *Proceedings of the ACM SIGMETRICS'98*, Madison, Wisconsin, USA, June 1998.

[10] "Iperf," http://dast.nlanr.net/Projects/Iperf/.

[11] J. C. Bolot, "Characterizing end-to-end packet delay and loss in the internet," in *Journal of High Speed Networks*, Atlanta, GA, Sept. 1993, p. 2(3):289297.

[12] C.-M. Cheng, H. Kung, and K.-S. Tan, "Use of spectral analysis in defense against DoS attacks," in *Proceedings of the IEEE GLOBECOM*, Taipei, China, 2002.