

# Spectral Models for Bitrate Measurement from Packet Sampled Traffic

Luigi Alfredo Grieco, *Member, IEEE*, Chadi Barakat, *Senior Member, IEEE*, and Michele Marzulli

**Abstract**—In network measurement systems, packet sampling techniques are usually adopted to reduce the overall amount of data to collect and process. Being based on a subset of packets, they introduce estimation errors that have to be properly counteracted by using a fine tuning of the sampling strategy and sophisticated inversion methods. This problem has been deeply investigated in the literature with particular attention to the statistical properties of packet sampling and to the recovery of the original network measurements. Herein, we propose a novel approach to predict the energy of the sampling error in the real time estimation of traffic bitrate, based on spectral analysis in the frequency domain. We start by demonstrating that the error introduced by packet sampling can be modeled as an aliasing effect in the frequency domain. Then, we derive closed-form expressions for the Signal-to-Noise Ratio (SNR) to predict the distortion of traffic bitrate estimates over time. The accuracy of the proposed SNR metric is validated by means of real packet traces. Furthermore, a comparison with respect to an analogous SNR expression derived using classic stochastic tools is proposed, showing that the frequency domain approach grants for a higher accuracy when traffic rate measurements are carried out at fine time granularity.

## I. INTRODUCTION

In network measurement systems, packet sampling techniques are usually adopted by network operators to reduce the overall amount of packets to capture and process [7], [8], [12]. They simply consist in capturing (randomly or periodically) a subset of packets, used to infer the original traffic properties. Obviously, such techniques introduce estimation errors that have to be properly counteracted with a fine tuning of the sampling strategy [18]. Given an upper limit on the tolerated load in routers and a set of targeted measurement tasks, there is always an appropriate sampling and inversion method<sup>1</sup> that allows the measurement to be carried out with the best possible accuracy. The mostly known sampling pattern consists of a random selection of packets at the incoming interfaces of routers with some predefined and homogeneous probability  $p$ . This probability  $p$  is called the *sampling rate* and is often set

This paper is an extended version of two previous communications at ACM IMC 2009 and IEEE Infocom 2010. It presents further experimental results and a comparison with a stochastic approach for the calculation of the Signal-to-Noise Ratio. It is partially funded by the European Commission through the FP7 ECOD project (INFSO-ICT-223936).

L. A. Grieco and M. Marzulli are with the DEE - Politecnico di Bari - Italy, e-mails: a.grieco@poliba.it and mic.marz@gmail.com

C. Barakat is with INRIA - Sophia Antipolis, France, e-mail: chadi.barakat@inria.fr

Manuscript submitted on 3-May-2010; revised on 8-Nov-2010, accepted on 23-Mar-2011. Associate Editor: J. Won-Ki Hong.

<sup>1</sup>In sampling terminology, *inversion* is the process of estimating original traffic properties from sampled measurements.

by operators to low values as 1/100 or even 1/1000 according to the bitrate of links, e.g., [4].

The performance of packet sampling has been deeply investigated in the literature with particular attention to the statistical properties of the sampled measurements and the way they should be inverted to recover the original traffic properties, e.g., [6], [9], [10], [13], [16], [21], [23]. Several metrics were studied as the traffic volume both in packets and bytes, the volume of the largest flows<sup>2</sup> often called heavy hitters, the number of flows, and the distribution of flow volumes. These previous works, among others, have shed light on many of the statistical properties of packet sampling. Several inversion methods have followed combining stochastic analysis and statistical inference. Most often, the aim of the inversion was to minimize the variance of the estimation error given a set of sampled packets collected during some time interval (i.e., fixed size population). However, in reality the traffic is not constant, but varies over time forming a signal composed of several frequencies. Packet sampling would then have different impact if studied over the entire signal duration rather than over some fixed time interval or a set of well defined packets. Instead of inverting a set of sampled packets with the minimum estimation error variance, we can ask the question of how to infer the spectrum of the original traffic with the minimum Signal-to-Noise Ratio (SNR). In this way, we can be sure that the main frequencies in the original traffic are preserved. This is of major importance for applications like anomaly detection, path characterization, traffic engineering and network tomography [4], [19], [20]. For example, in case of network anomalies, the anomalies, whether they are originated by attacks or failures, manifest themselves in the form of shifts in the bitrate of the traffic or a rapid change in the information it carries [27], [3]. These shifts map to some spectrum that needs to be correctly measured, and hence preserved, if someone wants the detection to be done in an efficient way. The same reasoning applies to traffic engineering where decisions on rerouting the traffic are taken by network administrators based on variations in the traffic bitrate.

In a recent communication [14], we had a look at packet sampling from the viewpoint of the spectral density of the original traffic. Our targeted measurement was the rate of the traffic (in packet/s or in byte/s) when tracked over time in some router interface and binned over fixed-size time windows that we denoted by  $T$ . We came up with a model for the traffic rate in the frequency domain that helped us

<sup>2</sup>A flow is a set of packets sharing common fields in their headers as the IP source address prefix, the IP destination address prefix and the port number.

in explaining the impact of packet sampling. As in classical signal theory, packet sampling was shown in [14] to introduce bias due to the replicas of the baseband component of the traffic signal. For network traffic with some well defined maximum cutoff frequency  $f_M$ , provided that the sampling rate  $p$  is sufficiently high, this bias can be eliminated by a proper low-pass filtering of the sampled traffic signal followed by an upscaling of the sampled signal by the inverse of the sampling rate. In particular, we show in [14] that to avoid aliasing effects, the inequality  $0.445/T < p/t_0 - f_M$  should hold, where  $t_0$  corresponds to the transmission time of the smallest packet over the monitored interface – or equivalently the minimum possible time between two consecutive packets. This theoretically ensures that both the inverted traffic rate and the real one possess the same spectral density when binned over time windows (with size  $T$ ). Unfortunately, the previous manipulation requires the existence of a maximum cutoff frequency in the network traffic, which might not be the case in reality as our measurements show. Moreover, outside the no aliasing region, the previous analysis does not give any idea on the energy of the noise introduced by packet sampling and on how much the inverted sampled traffic differs from the original one.

The previous limitation has been afforded in [15] by deriving closed-form expressions for the SNR that were able to predict the distortion of the traffic bitrate estimate. The proposed SNR models calculate for a given packet sampling probability  $p$  and an averaging time window  $T$ , the amount of error in each frequency band of the traffic rate signal, thus raising a trade-off between sampling overhead and frequency resolution. Indeed, for a fixed  $p$ , increasing  $T$  allows smaller estimation errors, at the expense of a coarser time resolution. On the opposite, to achieve a small estimation error using a small value of  $T$ , one needs very high values of  $p$ , with a consequent increase in the monitoring overhead. With our expressions of the SNR, network operators can tune their monitoring system, i.e.,  $T$  and  $p$ , in order to achieve the targeted accuracy at the desired time resolution. Moreover, being expressed in closed-form, our SNR models can be easily implemented in real network monitoring systems.

In the present paper, we make a further step ahead by extending preliminary results reported in [14], [15] in a more comprehensive framework supported by further experiments. Moreover, we compare the effectiveness of the proposed frequency-based approach with respect to a classical stochastic-based one, using the real packet traces collected by the MAWI project over Asian transpacific links [2] and the CAIDA association over OC 192 links [1]. Results show that the frequency-based approach is a very powerful tool, granting a high estimation accuracy in all operative conditions. Moreover, it outperforms the stochastic-based model for small values of the averaging time window  $T$ . This practically means that the proposed analysis is particularly attractive for monitoring applications requiring high time resolutions. A typical example is applications targeting the detection of network anomalies whether caused by malicious activities or failures of equipments or protocols, such as routing transitions, link outages, Denial-of-Service attacks, port scans, worms,

transient congestions, etc. An anomaly often manifests itself in the form of a rapid, and often short term, change of the network traffic either at the bitrate level or at the content level. This rapid change translates into high frequency components that any efficient monitoring system should be able to capture otherwise the anomaly might pass undetected. Our analysis provides the necessary instructions on how the sampling rate should be tuned in presence of these applications given the required time granularity.

The rest of the work is organized as follows. Section II formulates the problem of estimating the network traffic rate in the frequency domain. In Section III we explain how packet sampling introduces aliasing. Then we exploit this result in Section IV to derive SNR metrics that account for the impact of packet sampling on the spectral density of the traffic rate estimation. As reference, we also present an approximation of the SNR that uses stochastic analysis and that does not require the knowledge of the spectrum of the traffic rate. Section V validates and compares the proposed SNR metrics using real packet traces. Section VI overviews the related work and the last Section draws conclusions.

## II. PROBLEM FORMULATION IN THE FREQUENCY DOMAIN

We outline in this section our model for the analysis of packet sampling in the frequency domain. This model has been introduced for the first time in [14] and is exploited here to calculate the Signal-to-Noise Ratio (SNR) while inverting the sampled traffic. Our targeted measurement is the amount of data sent from a sender node ( $S$ ) to a receiver node ( $R$ ) averaged over time intervals of duration  $T$ , which will be referred to as bin or averaging time window. Our objective is to capture correctly the spectrum, and hence the amplitude and oscillations, of this time varying signal. In our analysis, a node can be a net or a subnet with some IP address prefix, a domain, an edge router, or even a single host. The estimation of the binned values of the traffic is carried out using packet sampling, i.e., each packet is captured or not with a uniform probability  $p$ , then the number of captured packets per bin is divided by the sampling rate  $p$  to infer the original rate of the traffic for that bin. The monitor moves then to estimate the rate of the traffic for the following bin and so on. To model the spectral density of the traffic signal, we divide the time axis into small time slots with size  $t_0$ , naturally smaller than  $T$ . In each slot, no more than one packet can be transmitted. In practice, this  $t_0$  corresponds to the transmission time of the smallest packet over the monitored link (or interface). One can also see it as the minimum possible time between two consecutive packets over the monitored link. Then, we define  $d(k)$  as the amount of data sent by  $S$  during the time interval  $[(k) \cdot t_0, (k+1) \cdot t_0)$ , where  $k \in N$ . To be more precise, if the transmission of an entire packet has been accomplished during the time interval  $[(k) \cdot t_0, (k+1) \cdot t_0)$ ,  $d(k)$  will be equal to the size of the sent packet, otherwise  $d(k)$  will be equal to 0. Moreover, we take the bin size  $T$  as an integer multiple of  $t_0$ , i.e.,  $T$  is made by  $T/t_0$  slots. The Fourier Transform of  $d(k)$

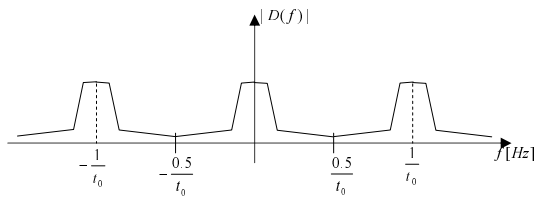


Fig. 1. Spectrum of original packet stream  $d(k)$ .

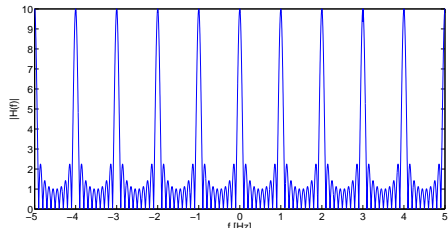


Fig. 2. Module of  $H(f)$  ( $t_0 = 1$  s and  $T = 10$  s).

can be expressed as follows [24]:

$$D(f) = \sum_{k=-\infty}^{+\infty} d_k \cdot e^{-j2\pi k f t_0} = \sum_{k=-\infty}^{+\infty} D_0(f - \frac{n}{t_0}), \quad (1)$$

where  $D_0(f) = 0$ , for  $|f| > \frac{0.5}{t_0}$ . This expression has a general validity because the spectrum of any discrete-time signal is periodic with period equal to  $1/t_0$ , if the time between two subsequent samples is equal to  $t_0$ . Basically,  $D_0(f)$  is a function that we introduce and that includes all frequencies of the signal  $d(k)$  in the interval  $[-0.5/t_0, +0.5/t_0]$ . To better clarify the meaning of our notation, Fig. 1 pictures a typical example of  $D(f)$ .

As first step, we model the spectrum of the traffic signal under the ideal assumption of capturing all packets, i.e.,  $p = 1$ . The spectrum of the sampled traffic is derived in the next section. Given that the measurement bin lasts  $T/t_0$  slots, summing the data received in a bin time can be seen as filtering  $d(k)$  using a discrete-time filter with pulse response  $h(k) = 1$  for  $k = 0 \dots \frac{T}{t_0} - 1$ , and  $h(k) = 0$  for  $k \geq \frac{T}{t_0}$ . The corresponding transfer function is:

$$H(f) = e^{-j\pi f (\frac{T}{t_0} - 1)t_0} \cdot \text{sin}(\pi f T) / \text{sin}(\pi f t_0). \quad (2)$$

$H(f)$  is a low-pass filter with cutoff bandwidth  $B \approx \frac{0.445}{T}$  and static gain equal to  $T/t_0$  [17]. Moreover, it is worth noting that the spectrum of  $H(f)$  is periodic (with period  $1/t_0$ ) since the corresponding pulse response is discrete. Thus,  $H(f)$  acts as a low-pass filter in the frequency band  $[-0.5/t_0, 0.5/t_0]$ .<sup>3</sup> To provide a further insight into the filter  $H(f)$ , Fig. 2 plots the module of its transfer function obtained for  $t_0 = 1$  s and  $T = 10$  s. Binning the traffic allows then to extract a filtered version  $\bar{d}(k)$  of the signal  $d(k)$ , using a low-pass filter with pulse response  $h(k)$ . Being  $H(f)$  a linear filter, it holds that

<sup>3</sup>Frequency components of  $d(k)$  outside the interval  $[-0.5/t_0, 0.5/t_0]$  can be filtered out only using an interpolator, i.e., a continuous time filter, that reconstructs a continuous version of the signal  $d(k)$ .

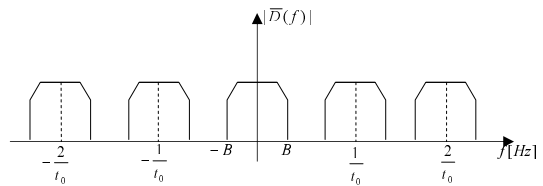


Fig. 3. Approximated model of  $\bar{D}(f)$ .

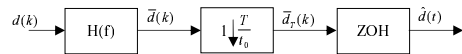


Fig. 4. Continuous time reconstruction of original packet stream  $d(k)$ .

the Fourier Transform of  $\bar{d}(k)$  is:

$$\bar{D}(f) = H(f)D(f) = \frac{T}{t_0} \sum_{n=-\infty}^{+\infty} \bar{D}_0(f - n/t_0), \quad (3)$$

where  $\bar{D}_0(f) = t_0 \frac{H(f)D_0(f)}{T}$ . The last equality in Eq. (3) holds because both  $H(f)$  and  $D(f)$  are periodic functions with the same period  $1/t_0$ . Fig. 3 plots an approximated model of  $\bar{D}(f)$ .

Now, we present our approach to move from a discrete-time signal representation to a continuous-time one. This is shown in Fig. 4: the signal  $\bar{d}(k)$  is decimated by a factor  $T/t_0$ , i.e., one sample of  $\bar{d}(k)$  is taken every bin, then the resulting signal  $\bar{d}_T(k)$  is processed with a Zero Order Holder (ZOH), which is a device that keeps the output  $\hat{d}(t)$  equal to the last received sample. Using the Poisson summation formula [24], the spectrum of  $\bar{d}_T(k)$ , i.e., the decimated version of  $\bar{d}(k)$ , is:

$$\bar{D}_T(f) = \sum_{n=-\infty}^{+\infty} \bar{D}_0(f - n/T). \quad (4)$$

It is worth noting that the spectrum  $\bar{D}_T(f)$  is the sum of the functions  $\bar{D}_0(f - \frac{n}{T})$ , which are obtained by translating  $\frac{T}{t_0} \cdot \bar{D}_0(f)$  by integer multiples of  $\frac{1}{T}$  and by dividing the result by  $T/t_0$ . As a consequence, and given that the bandwidth of  $\bar{D}_0(f)$  is  $B \approx \frac{0.445}{T}$  [17], the decimation does not introduce aliasing. Moreover, the transfer function of the ZOH is:

$$G_{ZOH}(f) = e^{-j\pi f T} \cdot \text{sin}(\pi f T) / (\pi f T), \quad (5)$$

which is a low-pass filter with unitary static gain and bandwidth equal to that of  $H(f)$ . With respect to  $H(f)$ , the ZOH is also able to filter out all high frequency components of the input signal, so that, the spectrum of the continuous-time signal  $\hat{d}(t)$  is no more periodic and can be expressed as follows:

$$\hat{D}(f) = G_{ZOH}(f)\bar{D}_T(f) \approx G_{ZOH}(f)\bar{D}_0(f). \quad (6)$$

This is no other than a low-pass filtered version of the baseband component of the spectrum of  $d(k)$ . The signal  $\hat{d}(t)$  is what network operators track over time on their router interfaces. Our aim is to evaluate the impact of packet sampling on the spectrum of this signal and hence to propose conservative values for  $T$  and  $p$  to be used. Such values should ensure that the estimated binned traffic and the real binned traffic do not differ from each other more than the error margin defined by

the operator. Note that most of the difficulty comes from the fact that the spectrum of the original signal  $d(k)$  is unknown from sampled traffic, so one has to estimate it jointly with the optimization of  $p$  and  $T$ .

### III. SPECTRUM OF SAMPLED TRAFFIC

Following the same methodology, we derive the spectrum of the binned traffic rate estimated from sampled packets. We show that: (i) this spectrum is an aliased version of the spectrum of the original traffic; (ii) the impact of aliasing grows by lowering  $p$ ; (iii) the energy of the noise due to aliasing can be reduced by increasing the time bin  $T$ .

Suppose that packets are sampled with some uniform probability  $0 < p < 1$  and denote by  $d_p(k)$  the volume of sampled data in the time slot  $[(k) \cdot t_0, (k+1) \cdot t_0)$ ,  $k \in \mathbb{N}$ . Given that no more than one packet can appear in a tinny time slot  $t_0$ , the signals  $d(k)$  and  $d_p(k)$  are related to each other, as for each  $k$ ,  $d_p(k)$  is equal to  $d(k)$  with probability  $p$  and to 0 with probability  $1 - p$ . Let us express the time slot corresponding to the  $n$ -th captured sample of  $d(k)$  as  $t_n = (\frac{n}{p} + \Delta_n)t_0$ ,  $\Delta_n$  being a random variable modeling the variability of the time between sampled packets. Under this hypothesis, we can compute the spectrum of  $d_p(k)$  as:

$$D_p(f) = \sum_{n=-\infty}^{+\infty} d(\frac{n}{p} + \Delta_n) e^{-j2\pi f(\frac{n}{p} + \Delta_n)t_0} \quad (7)$$

$$= \sum_{n=-\infty}^{+\infty} d(\frac{n}{p} + \Delta_n) e^{-j2\pi f \frac{n}{p} t_0} \left( 1 + \sum_{i=1}^{+\infty} \frac{(-j2\pi f \Delta_n t_0)^i}{i!} \right). \quad (8)$$

Since we are interested in low-frequency components, with  $|f| < \frac{1}{T}$ , we can safely assume that  $f\Delta_n t_0 \ll 1$  or equivalently  $\Delta_n \ll T/t_0$ . This simply means that the time bin size is much larger than the jitter of the inter-arrival time between sampled packets. Thus:

$$D_p(f) \approx \sum_{n=-\infty}^{+\infty} d(n/p + \Delta_n) e^{-j2\pi f \frac{n}{p} t_0}. \quad (9)$$

Assuming further that  $d(\frac{n}{p} + \Delta_n) = d(\frac{n}{p}) + e_n$ , where  $e_n$  is a zero mean signal containing high frequencies components only (i.e., low frequency components are captured by  $d(\frac{n}{p})$ ), we can express  $D_p(f)$  as follows:

$$D_p(f) \approx \sum_{n=-\infty}^{+\infty} d(n/p) e^{-j2\pi f \frac{n}{p} t_0} + \sum_{n=-\infty}^{+\infty} e_n e^{-j2\pi f \frac{n}{p} t_0}. \quad (10)$$

Given that in the frequency band of interest  $e_n$  has a negligible energy, the spectrum of the sampled traffic  $D_p(f)$  can be viewed as the spectrum of the original traffic  $d_k$  subsampled with frequency  $\frac{p}{t_0}$ . Recalling the spectrum of the signal  $d_k$  reported in Eq. (1), it holds that [24]:

$$D_p(f) \approx p \sum_{n=-\infty}^{+\infty} D_0(f - n \cdot p/t_0). \quad (11)$$

An example of this spectrum is plotted in Fig. 5 where we can see the aliasing introduced by packet sampling. The effect of the aliasing cannot be fully filtered out given the overlap of baseband replicas. However, the amount of noise due to aliasing can be reduced by low-pass filtering  $d_p(k)$ .

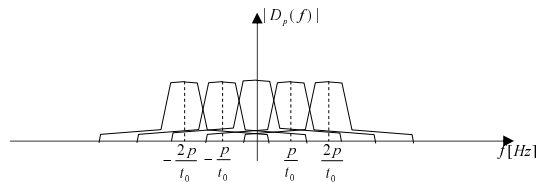


Fig. 5. Spectrum of sampled packet stream  $d_p(k)$ .

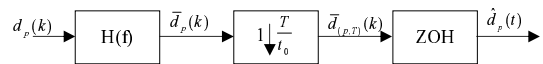


Fig. 6. Continuous time reconstruction of sampled packet stream  $d_p(k)$ .

That is exactly what the binning of the sampled traffic ensures, with  $H(f)$  defined in Eq. (2) being the transfer function of the resulting low-pass filter. Thus, after packet sampling, the reduced traffic  $d_p(k)$  is filtered using  $H(f)$  to obtain the signal  $\bar{d}_p(k)$ . Its spectrum can be expressed as:

$$\bar{D}_p(f) \approx p H(f) \sum_{n=-\infty}^{+\infty} D_0(f - n \cdot p/t_0). \quad (12)$$

By isolating the baseband component  $\bar{D}_0(f)$ , this can be rewritten as:

$$\bar{D}_p(f) \approx p \frac{T}{t_0} \bar{D}_0(f) + p H(f) \sum_{n \neq 0} D_0(f - n \cdot p/t_0) \quad (13)$$

Finally, in order to move to a continuous time representation that models the averaging of the sampled traffic over bins of  $T/t_0$  slots, the signal  $\bar{d}_p(k)$  has to be decimated by a factor  $T/t_0$  before being interpolated using a ZOH (see Fig. 6). Using the Poisson summation formula as done to derive Eq. (4), and provided that the filter  $H(f)$  has made negligible the aliasing due to the sampling, the spectrum of  $\bar{d}_{(p,T)}(k)$ , i.e., the decimated version of  $\bar{d}_p(k)$ , can be written as:

$$\bar{D}_{(p,T)}(f) \approx p \sum_{n=-\infty}^{+\infty} \bar{D}_0(f - n/T). \quad (14)$$

Next, by applying the ZOH, one can extract a continuous time reconstruction of the sampled traffic whose spectrum is  $p G_{ZOH}(f) \bar{D}_0(f)$ , i.e., a low-pass filtered version of the baseband component of the average spectrum of  $d(k)$  scaled by  $p$ . Compared to Eq (6), this confirms indeed that the signal  $d_p(k)$  modeling the sampled traffic has to be divided by  $p$  to compensate the scaling due to the aliasing and to obtain the same spectrum as the time averaged reconstruction of the original traffic. From now on, we will always consider the inverted signal  $d_p(k)/p$  in our analysis.

### IV. SIGNAL-TO-NOISE RATIO

The previous analysis relates the spectrum of the packet sampled traffic to the one of the original traffic. Packet sampling introduces aliasing caused by the replicas of the main component that overlap inside the frequency band of interest, see Fig. 5. The amount of aliasing depends on several factors: the spectrum of the original traffic modeled by  $D_0(f)$ , the sampling rate  $p$ , and the time bin  $T$  used to

average measurements. Theoretically, and if the original traffic only presents low frequency components (i.e., a small cutoff frequency  $f_M$ ), this aliasing can be removed by an appropriate low pass filtering, at a condition that the sampling rate is not very small so that the inequality  $0.445/T < p/t_0 - f_M$  is satisfied. In practice, and as stated in the Introduction, such manipulation is not possible given the absence of a clear cutoff frequency  $f_M$  in the spectrum of the original traffic (see Fig. 7 (a) for an example of a real spectrum in its baseband). Aliasing hence exists for all frequency components. The only solution we are left with is to limit the noise introduced by aliasing by appropriately tuning the sampling rate  $p$  and the averaging measurement bin  $T$ .

In this section we propose robust metrics to evaluate the reliability of the traffic rate estimation from sampled traffic. Our metrics are function of the sampling rate  $p$ , the bin size  $T$ , the probability to find a busy slot in the original traffic  $p_L$ , i.e.,  $p_L = P(d(k) > 0)$ , and the first and second order moments of the packet size, which will be referred to as  $\alpha$  and  $\beta$ , respectively. All these parameters can be calculated from the sampled traffic without access to the original traffic, hence the interest of our approach. To clarify this approach, we will first derive an expression for the SNR assuming that all packets have the same size. Then, we will extend this finding to the most general case of packets with different sizes. The effectiveness of both  $SNR$  expressions will be proved using real packet traces. They will be computed by exploiting the theoretical findings presented in the previous sections. Moreover, the analysis will be made even more realistic by using the Discrete Fourier Transform (DFT) [24], which works with finite sets of collected packets. The regular Fourier Transform from its side requires an infinite set to be estimated. For sake of clarity, Tab. I lists all parameters used in our theoretical derivations.

TABLE I  
MAIN MODEL PARAMETERS.

Parameter	Meaning
$p$	packet sampling probability
$T$	bin size
$t_0$	time-slot size
$f_M$	cut-off frequency
$N$	total number of time-slots in a trace
$\alpha$	first order moment of packet size
$\beta$	second order moment of packet size
$p_L$	probability to find a busy slot
$C$	average packet transmission rate

#### A. SNR under a Constant Packet Size

Given a discrete signal  $d(k), k = 0 \dots N - 1$ , modeling packet sizes across  $N$  time slots, its DFT coefficients  $D_{dft}(n), n = 0 \dots N - 1$ , can be expressed as follows:

$$D_{dft}(n) = \sum_{k=0}^{N-1} d(k) e^{-\frac{2\pi j}{N} kn}, n = 0 \dots N - 1. \quad (15)$$

In this analysis we consider that all packets have the same size  $d_0$ , an assumption that we relax later. This implies that  $d(n)$

can be either equal to  $d_0$  or 0. Without loss of generality, we take  $d_0 = 1$ .<sup>4</sup> From the Parseval theorem, we can write that:

$$\sum_{k=0}^{N-1} d^2(k) = \frac{1}{N} \sum_{n=0}^{N-1} |D_{dft}(n)|^2. \quad (16)$$

The summation on the left-hand side of the equation is no other than the energy carried by the original packet stream. It follows that the  $n$ th DFT coefficient carries an amount of energy equal to  $|D_{dft}(n)|^2/N$ . On the other hand, one can write the total energy of the signal in this particular case of packet of size equal to one as:

$$E^T = \sum_{k=0}^{N-1} d^2(k) = p_L \cdot N. \quad (17)$$

Recalling Eq. (15), we can write that  $D_{dft}(0) = p_L \cdot N$ . As a consequence, the coefficient  $D_{dft}(0)$ , which models the continuous component of the traffic signal, carries an amount of energy equal to  $|D_{dft}(0)|^2/N = p_L E^T$ .

Motivated by real traffic spectrum measurements (as the ones reported in Fig. 7), we suppose that the remaining energy of the signal  $(1 - p_L)E^T$  is uniformly spread over the  $D_{dft}(n)$  coefficients having  $1 \leq n < N$ . Thus, we can derive that the amount of energy carried by the generic  $D_{dft}(n), 1 \leq n < N$ , is equal to  $p_L(1 - p_L) \frac{N}{N-1} \approx p_L(1 - p_L)$ , for sufficiently large values of  $N$ . Given this spectrum description, it follows that the energy of the signal  $\bar{d}$ , i.e., a filtered version of  $d$  obtained using a low-pass filter with a bilateral bandwidth equal to  $\frac{N_B}{N t_0}$ , is given by:

$$E^S = p_L^2 N + p_L(1 - p_L)N_B. \quad (18)$$

$N_B$  can be seen as the number of frequency components around the continuous one that are allowed by the low-pass filter. In a similar way, we can write the energy of the tail of  $d$  comprised in a bandwidth of size  $\frac{N_B}{N t_0}$  as:

$$E_{tail} = p_L(1 - p_L)N_B. \quad (19)$$

Now, we have to recall that the noise in the sampled traffic signal  $d_p$  caused by aliasing is due to the tails of the spectrum of the original traffic signal  $d$  translated and folded together in the bandwidth of interest. If the sampling probability is  $p$ , we expect to have a number of replicas equal to  $\frac{1-p}{p}$ . Each replica is scaled down by  $p$  because of sampling as stated by Eq. (14). To compensate for this, we amplify the signal  $d_p$  by  $1/p$ . The energy of the noise  $E^N$  becomes:

$$E^N = \frac{1-p}{p} E_{tail} = \frac{1-p}{p} p_L(1 - p_L)N_B. \quad (20)$$

This is the SNR value we are looking for can be computed as follows:

$$SNR = \frac{E^S}{E^N} = \frac{p_L^2 N + p_L(1 - p_L)N_B}{\frac{1-p}{p} p_L(1 - p_L)N_B}. \quad (21)$$

This metric is a function of  $p_L$ , which can be accurately estimated by dividing the number of sampled packets by  $pN$ .

<sup>4</sup>We are computing a ratio between the energy of the signal and the energy of the noise: any value assigned to  $d_0$  would disappear from our expressions.

As for  $N_B$ , it is related to the time bin over which the traffic is monitored. A binning over  $T$  seconds is equivalent to filtering the traffic signal using a low-pass filter of bilateral bandwidth equal to  $\frac{0.89}{T} = \frac{N_B}{Nt_0}$ . By replacing  $N_B$  by its expression as a function of  $T$ , we get the final expression for the SNR metric for constant packet sizes:

$$SNR = \frac{p_L + (1 - p_L)0.89t_0/T}{\frac{1-p}{p}(1 - p_L)0.89t_0/T}. \quad (22)$$

### B. SNR in a realistic scenario

Here, we relax the assumption of having a constant packet size in order to provide a SNR metric that better reflects realistic scenarios. Our main finding can be summarized as follows:

*Proposition 1: Let  $\alpha$  and  $\beta$  be respectively the first and second order moments of the packet size. Let  $p_L$  be the probability to find a busy slot in the original traffic, i.e.,  $p_L = P(d(k) > 0)$ ,  $p$  the sampling rate, and  $T$  the time bin length. The Signal-to-Noise Ratio caused by sampling can be approximated by*

$$SNR = \frac{p_L\alpha^2 + (\beta - p_L\alpha^2)0.89t_0/T}{\frac{1-p}{p}(\beta - p_L\alpha^2)0.89t_0/T}. \quad (23)$$

*Proof:* In this case of variable packet sizes, the value  $q$  of the total energy carried by the original traffic can be defined and estimated as follows:

$$q = E^T = \sum_{k=0}^{N-1} d^2(k) \approx N \cdot p_L \cdot \beta. \quad (24)$$

Moreover, the value for  $D_{df_t}(0)$  can be estimated as:

$$D_{df_t}(0) = \sum_{k=0}^{N-1} d(k) \approx N \cdot p_L \cdot \alpha. \quad (25)$$

This gives the following approximation for the energy  $t$  associated to the first DFT coefficient:

$$t = \frac{|D_{df_t}(0)|^2}{N} \approx N \cdot (p_L \cdot \alpha)^2. \quad (26)$$

Now, assuming that the remaining energy of the signal  $q - t$  is uniformly spread over the other  $D_{df_t}(n)$  coefficients having  $1 \leq n < N$ , we can derive that the amount of energy carried by the generic  $D_{df_t}(n)$ ,  $1 \leq n < N$ , is equal to  $\frac{q-t}{N-1}$ . The energy of the signal  $\bar{d}$ , i.e., the filtered version of  $d$  obtained using a low-pass filter with a bilateral bandwidth equal to  $\frac{N_B}{Nt_0}$ , becomes equal to:

$$E^S = t + \frac{q-t}{N-1}N_B. \quad (27)$$

Moreover, the energy of the tail of  $d$  comprised in a bandwidth of size  $\frac{N_B}{Nt_0}$  is equal to:

$$E_{tail} = \frac{q-t}{N-1}N_B. \quad (28)$$

As for the constant packet size case, the noise in the signal  $d_p$  caused by aliasing is due to the tail of the spectrum of the signal  $d$  translated and folded together in the bandwidth of interest. Having in total  $\frac{1-p}{p}$  replicas that overlap with

the baseband component, the energy of the noise  $E^N$  can be written as follows:

$$E^N = \frac{1-p}{p}E_{tail} = \frac{1-p}{p}\frac{q-t}{N-1}N_B. \quad (29)$$

This gives the following expression for the SNR:

$$SNR = \frac{E^S}{E^N} = \frac{t + \frac{q-t}{N-1}N_B}{\frac{1-p}{p}\frac{q-t}{N-1}N_B}. \quad (30)$$

By substituting Eqs. (24) and (26) in (30), considering that  $N/(N-1) \approx 1$  for a sufficiently large  $N$ , and setting the bilateral bandwidth of the low-pass filter to  $\frac{N_B}{Nt_0} = \frac{0.89}{T}$  for time bins of length  $T$ , one can find the result (23) stated in the proposition. This concludes the proof.  $\square$

### C. Asymptotic SNR

Both Eqs. (22) and (23) have been derived assuming a given  $t_0$ , modeling the smallest packet transmission time. With the increase in the link speed, this time is in practice very small<sup>5</sup> giving sense to an asymptotic version of the above expressions (22) and (23), obtained by letting  $t_0$  tend to zero. It can be easily shown that, as  $t_0$  tends to zero, Eq. (22) becomes:

$$SNR = \frac{p(T \cdot C + 0.89)}{(1-p)0.89}, \quad (31)$$

whereas Eq. (23) becomes:

$$SNR = \frac{p(T \cdot C\alpha^2 + \beta 0.89)}{(1-p)0.89\beta} = \frac{p}{1-p} \left[ \frac{TC\alpha^2}{0.89\beta} + 1 \right], \quad (32)$$

$C$  being the average packet transmission rate (total number of packets over observation time). The demonstration follows by substituting  $p_L = Ct_0$  in Eqs. (22) and (23) and then letting  $t_0$  tend to 0.

### D. Simple analysis of the SNR using stochastic analysis

To provide a ground for comparison, we derive hereafter a simple estimator for the Signal-to-Noise Ratio using stochastic analysis. The analysis is on average over all sampling realizations. Contrary to the previous analysis, the details of the spectrum are ignored, only distributions and moments are kept. This analysis, simple to carry out, is meant to illustrate the benefits of the spectral analysis and to point to scenarios where it outperforms its stochastic counterpart.

Let  $d_i(1), d_i(2), \dots, d_i(n_i)$  be the sequence of packet sizes during time interval  $T_i$ ,  $n_i$  being the number of these packets. Our target signal is no other than the sum of this sequence, for different  $i$ . Denote this sum by  $X_i = \sum_{k=1}^{n_i} d_i(k)$ . Let  $\hat{X}_i$  be the estimator of this sum from the sampled traffic. The estimation is done by summing the sizes of sampled packets, then dividing this sum by the sampling rate  $p$ . It is straightforward to say that the estimator of  $X_i$  is unbiased, which means that the expectation of  $\hat{X}_i$  over all sampling realizations is simply  $X_i$ . We calculate the energy of the error

<sup>5</sup>The tiny time slot  $t_0$  is equal to  $3.2 \mu\text{s}$  in a 100Mbps link with a minimum packet size equal to 40 bytes (case of an ACK).

of this estimator during time interval  $T_i$ , which is no other than the variance of the estimator  $\hat{X}_i$ , summed over  $T_i$ .

$$E_i = T \cdot \text{VAR}(\hat{X} - X) = T \cdot \text{VAR}(\hat{X}) = \frac{1-p}{p} T \sum_{k=1}^{n_i} d_i^2(k). \quad (33)$$

If  $N$  is the total number of packets in the trace, and  $L$  its total duration, then the total energy of the error in the trace is simply  $\sum_i E_i = \frac{(1-p)}{p} T \sum_{k=1}^N d_k^2(k)$ . With  $\beta$  denoting the second order moment of the packet size, this total energy of error per trace can be approximately written as  $\frac{(1-p)}{p} NT\beta$ .

We now provide an approximation for the energy of the original binned traffic rate. Unfortunately, this energy is unknown and cannot be directly measured, since the traffic itself is already sampled and its original version is not available. One needs to make an assumption on the shape of the original traffic to be able to approximate this energy. We make the simple and conservative assumption of constant traffic rate equal to  $TN\alpha/L$ , where  $\alpha$  as defined before, is the first moment of the packet size. The total energy of the traffic rate over all the trace is simply equal to  $(TN\alpha)^2/L$ . It follows that a simple expression of the SNR is,

$$\text{SNR} = \frac{p}{1-p} \frac{NT}{L} \frac{\alpha^2}{\beta} = \frac{p}{1-p} CT \frac{\alpha^2}{\beta}. \quad (34)$$

We recall that  $C$  is the average rate of packets in the entire trace. Note that this expression does not account for the oscillations of the traffic. It supposes that the traffic is constant. Our spectral model makes a more advanced representation of the traffic and its spectrum. Also note in this expression how the SNR increases linearly with  $T$ , which sounds logic since when the binning window becomes larger and larger, the effect of sampling becomes less and less. Also note how this expression increases when increasing the sampling rate and decreases (more noise) when packet sizes become more variable.

A further point to consider is the analytical comparison with the SNR expressions derived above using the frequency domain analysis. We provide here a simple comparison with the approximated SNR in Eq. (32), which also depends on  $T$ ,  $\alpha^2$ , and  $\beta$  as Eq. (34) does. Further comparison is done later in the paper by the help of experimentations on real traces. The added value of the expression in Eq. (32) is that it models the traffic spectrum and captures the low-pass filtering effect of the binning. This mainly translates into the coefficient 0.89 in the expression. In fact, if we neglect the second addend in the squared brackets of Eq. (32), we find that the SNR computed using statistic analysis is  $1/0.89 = 1.12$  times smaller than the one that can be obtained using the frequency domain analysis. This implies that, if the expressions we derived using the frequency domain approach are able to fit real packet traces (this will be demonstrated in the next section), the simple statistical approach will underestimate the real SNR by a multiplicative factor. The experimental validation will confirm this observation. It is also worth noting by comparing Eqs. (32) and (34), that the frequency model requires just one sum and one multiplication more than the stochastic one, i.e., both models have similar computational complexities.

## V. EXPERIMENTAL RESULTS

### A. Spectrum of a sampled traffic trace

We start by giving an example on the spectrum of the network traffic in its baseband and on how this spectrum becomes after packet sampling. To this aim, we process a 15 minutes long real packet trace collected in January 2009 over a trans-pacific 150 Mbps link by the Japanese MAWI project<sup>6</sup> [2]. Fig. 7 shows the module of the spectrum of the inverted sampled traffic  $d_p(k)/p$ , obtained for several values of  $p$ , when the first 10 thousand packets of the trace are considered. First the trace is sampled, then Eq. (1) is used for spectrum evaluation; the summation is done over all time slots  $t_0$  composing the trace. The time slot is set to the minimum packet size available in the trace divided by the link speed, and sampled packets are assigned to time slots using their timestamps. By comparing the plot obtained for  $p = 1$  with the other ones, it is straightforward to note that: (i) only low frequencies of the original traffic can be recovered, even using a very high sampling probability as  $p = 0.1$ ; (ii) the harmonic tones of the original traffic, i.e., those obtained for  $p = 1$ , appear translated in the frequency spectrum of the sampled traffic signals as expected by the Poisson summation formula; and (iii) the noise across the continuous component of the traffic signal grows with  $1/p$  as expected by Eqs. (22) and (23).

### B. Root Mean Squared Relative Error

Herein, we validate the effectiveness of the SNR metrics proposed in Eqs. (22), (23), and (34), which will be referred to in the sequel as SPEC-CP (spectral model - constant packet size), SPEC-VP (spectral model - variable packet size), and STOCH (stochastic model), respectively. For this validation, we analyze five distinct non-sampled packet traces, each one lasting 15 minutes. The characteristics of the five traces are summarized in Table II. The first three ones have been obtained from the MAWI project [2] and were collected at two trans-pacific 150 Mbps links during January 2009 and December 2005<sup>7</sup>. The last two traces have been made available by the Cooperative Association for Internet Data Analysis (CAIDA) [1]. They were collected at two different OC 192 links (with almost 10 Gbps capacity) during November and December 2009. We sample the traces with probabilities ranging in the interval  $[10^{-4}, 1]$ . For each sampling probability, five distinct experiments are repeated using different seeds for the random number generator. Sampled traces are then averaged over time bins of length  $T$ , ranging in the interval  $[1 \text{ s}, 100 \text{ s}]$ . After verifying that there are enough sampled packets to calculate the spectrum and the different model parameters, we compare the measured SNR with respect to the estimated ones, using the three models SPEC-CP, SPEC-VP, and STOCH. For all experiments, we plot the square root of  $1/\text{SNR}$  as a function of the averaging time bin  $T$  for three distinct values of  $p$ , i.e.,  $p = 0.1$ ,  $p = 0.01$ , and  $p = 0.001$ , representing

<sup>6</sup>The trace is available at <http://mawi.wide.ad.jp/mawi/samplepoint-F/2009/>

<sup>7</sup>The traces are available at <http://mawi.wide.ad.jp/mawi/samplepoint-F/2009/> and <http://tracer.csl.sony.co.jp/mawi/samplepoint-B/2006/>

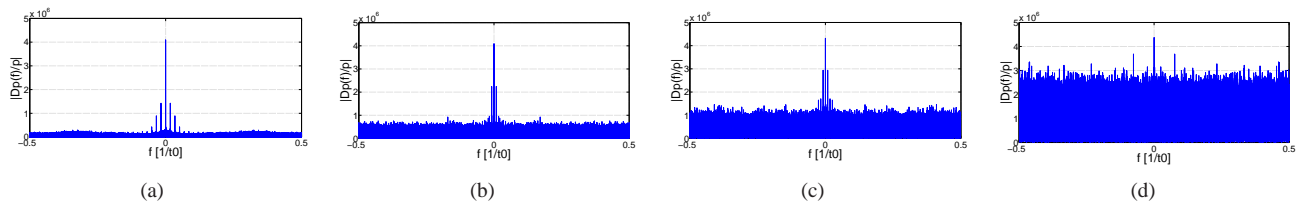


Fig. 7. Baseband component of  $D_p(f)/p$ : (a)  $p = 1$ ; (b)  $p = 0.1$ ; (c)  $p = 0.03$ ; (d)  $p = 0.005$ .

high, average, and low sampling probability, respectively. The  $\sqrt{1/SNR}$  metric captures the root mean squared relative error (RMSRE). Note that to change the traffic granularity, we process both aggregated traces and all sub-flows extracted from them and having the same first 8 bits of the sending IP address (often called the /8 IP source flows).

To better illustrate the experimental results, we split them into two parts. In the first part, we compare the three proposed models using the five aggregated traces, whereas, in the second part, we consider sub-flows extracted from each trace. The second part allows us to understand how the proposed models perform when we zoom into the traffic and increase the granularity of the bitrate measurement.

1) *Results for aggregate traces:* Results related to aggregate trace 1 are shown in Figs. 8 where the accuracy of the three considered models in estimating the RMSRE is reported. In particular, in Fig. 8 the estimated and measured RMSRE are plotted for three possible values of  $p$  by letting  $T$  vary from 1 s to 100 s. These results demonstrate that all considered models are able to capture the effect of packet sampling on the aggregate trace 1. Analogous results have been obtained also for traces 2 and 3 (see Figs. 9 - 10), even with some minor differences from model to model. In particular, we see that for traces 2 and 3, models SPEC-VP and STOCH provide a slightly better accuracy than model SPEC-CP. This effect can be explained by noting that for traces 2 and 3, the  $\alpha^2/\beta$  ratio is smaller than for trace 1 (see Table II), i.e., traces 2 and 3 have a higher packet size variability than trace 1, which cannot be captured by model SPEC-CP. This latter model assumes a constant packet size, whereas the two other models account for the variability of the packet size.

The accuracy in estimating the RMSRE worsens, however, for all the three models when traces 4 and 5 are processed (see Figs. 11-12). Both traces 4 and 5 are taken from links with a very low utilization (below 2%, see Table II). For such low utilizations (i.e., low  $p_l$  values) and particularly when  $T$  is small, all considered models slightly overestimate the RMSRE. Low utilizations are often the indication of high traffic variability compared to the average traffic rate (less multiplexing). All three models make assumptions about the variable traffic component: SPEC-CP and SPEC-VP assume that the spectrum has equal energy at frequencies other than the constant component; STOCH only accounts for the constant component and ignores the variable component. These assumptions are slightly violated when the traffic is highly variable leading to a larger estimation error. Note that surprisingly the SPEC-CP model provides better estimation of the SNR than the two other models. This is because Eq. (22)

provides higher SNR than Eqs. (23) and (34), i.e., smaller RMSRE, and, as a consequence the overestimation effect is mitigated.

From this first analysis we can conclude that: (i) the accuracy of considered models increases with the link utilization and consequently with  $p_l$ ; (ii) at sufficiently high  $p_l$ , as the packet size variability increases, the model SPEC-CP provides a smaller accuracy than models SPEC-VP and STOCH, which explicitly account for packets with different sizes; (iii) at low  $p_l$  and small  $T$  all models overestimate the RMSRE, but, for model SPEC-CP this effect is less evident because it provides higher SNR than the other considered models, thus counteracting the overestimation.

It is worth to note that the RMSRE observed in these experiments never exceeds 100%, even for small values of  $p$  and  $T$ . This is because the total number of packets of an aggregate trace is so high that packet sampling effects are mitigated. Furthermore, for the five considered traces, the maximum difference between measured and estimated RMSRE ranges from 3% to 40% as  $p$  decreases from 0.1 to 0.001. This roughly means an estimation error of the sampling noise that does not exceed 40% of the signal itself even in the most challenging conditions (very small  $p$  and  $T$ ).

TABLE II  
MAIN TRAFFIC PARAMETERS OF AGGREGATE TRACES

	Link Capacity (Mbps)	Link Usage (%)	$p_l$	$\alpha$ (Byte)	$\beta$ (Byte <sup>2</sup> )	$\alpha^2/\beta$
Trace 1 (MAWI) Jan. 2009	150	6	0.032	78	11317	0.54
Trace 2 (MAWI) Jan. 2009	150	13	0.015	341	400452	0.29
Trace 3 (MAWI) Dec. 2005	150	34	0.022	621	829127	0.46
Trace 4 (CAIDA) Nov. 2009	9953	0.3	0.0002	618	797477	0.48
Trace 5 (CAIDA) Nov. 2009	9953	1.1	0.0009	510	660227	0.39

2) *Results for sub-flows:* Given that the considered models are very sensitive to  $p_l$  (the probability of busy slot or differently speaking the link load), we test the three models with a finer traffic granularity to verify whether the aforementioned conclusions are still confirmed. Results regarding /8 IP source sub-flows extracted from trace 1 are reported in Fig. 13. We report herein both the average of all estimated RMSRE values

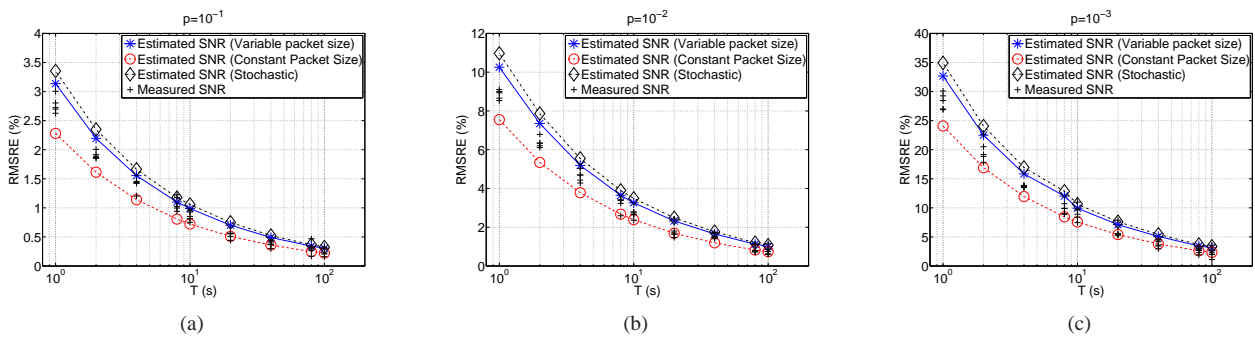


Fig. 8. RMSRE for aggregate trace 1 sampled at: (a)  $p = 0.1$ ; (b)  $p = 0.01$ ; (c)  $p = 0.001$ .

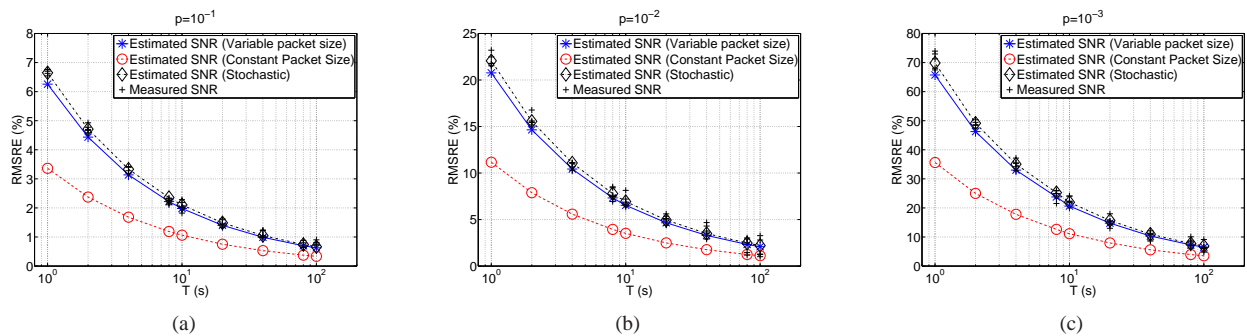


Fig. 9. RMSRE for aggregate trace 2 sampled at: (a)  $p = 0.1$ ; (b)  $p = 0.01$ ; (c)  $p = 0.001$ .

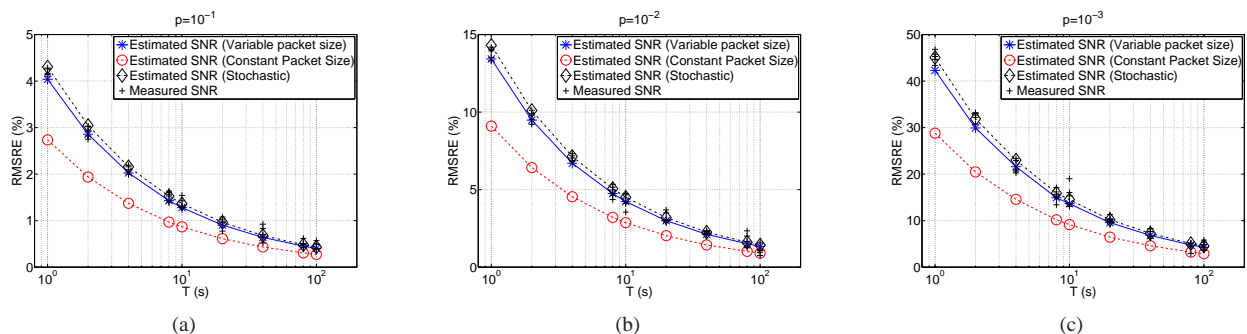


Fig. 10. RMSRE for aggregate trace 3 sampled at: (a)  $p = 0.1$ ; (b)  $p = 0.01$ ; (c)  $p = 0.001$ .

(for each  $p$  and  $T$ ) as well as measured RMSRE (error bars for the measured RMSRE represents the standard deviation over all sub-flows and sampling runs with the mean RMSRE in the middle of the bars)<sup>8</sup>. From these figures it is important to note that as long as the averaging bin  $T$  increases, the RMSRE decreases as well and all considered models converge towards the measured RMSRE. On the opposite, for values of  $T$  in the interval  $[1 \text{ s}, 10 \text{ s}]$ , the RMSRE is very high, and, under this condition, the stochastic model fully misses the measured values. This effect is particularly evident in Fig. 13 (a) and is confirmed also in Figs. 14 - 17, which report results obtained on sub-flows from traces 2-5. In fact the stochastic approach we have used assumes a constant-bit rate model for the traffic and hence ignores any variability of the traffic volume along the time window  $T$ . Due to this assumption, the energy of high

frequency components of the traffic is not taken into account. For low values of  $T$ , these high frequency components of the traffic become relevant making the mismatch between the STOCH model and the reality noticeable. Furthermore, the model SPEC-CP, computed assuming a constant packet size, is more accurate than SPEC-VP and STOCH in these scenarios. This effect can be explained by recalling that we are processing sub-flows, which have a value of  $p_i$  much smaller than the one for the aggregated traces. In these conditions, as already explained in the previous sub-section, all models overestimate the RMSRE at low  $T$ , but, for model SPEC-CP this effect is less evident because it provides higher SNR than other considered models, reducing the impact of the overestimation.

To conclude, we can say that models SPEC-VP and SPEC-CP are robust enough to allow an operator to properly tune  $p$  and  $T$ , in several operating conditions, depending on the goals of the monitoring systems. The SPEC-CP model has the further advantage of compensating the estimation error in case of low rate traffic.

<sup>8</sup>We avoid to plot confidence intervals for estimated RMSRE to improve the readability of plots. Error bars for estimated RMSRE are very similar to those of measured RMSRE.

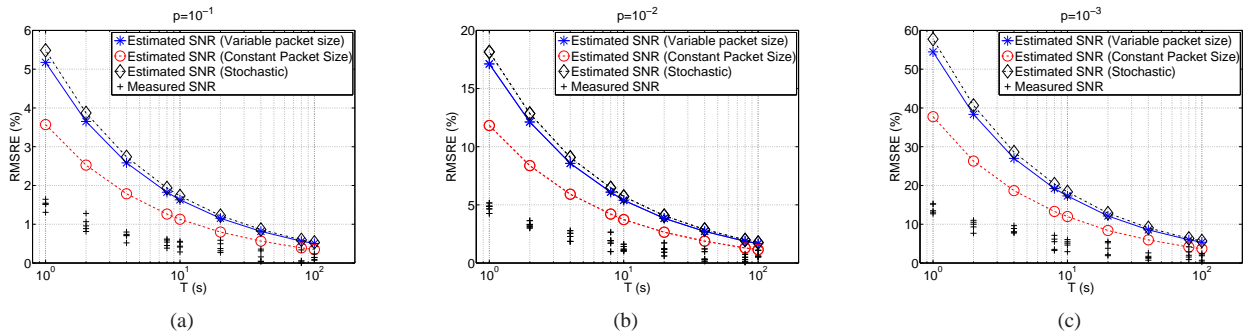


Fig. 11. RMSRE for aggregate trace 4 sampled at: (a)  $p = 0.1$ ; (b)  $p = 0.01$ ; (c)  $p = 0.001$ .

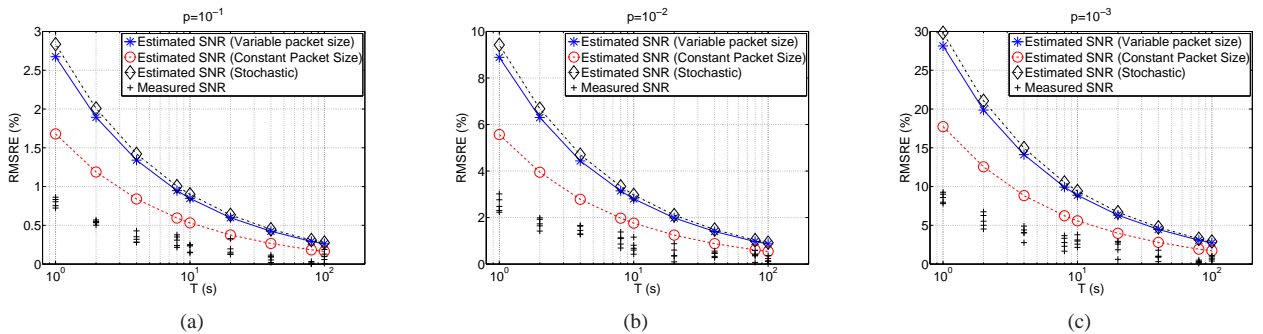


Fig. 12. RMSRE for aggregate trace 5 sampled at: (a)  $p = 0.1$ ; (b)  $p = 0.01$ ; (c)  $p = 0.001$ .

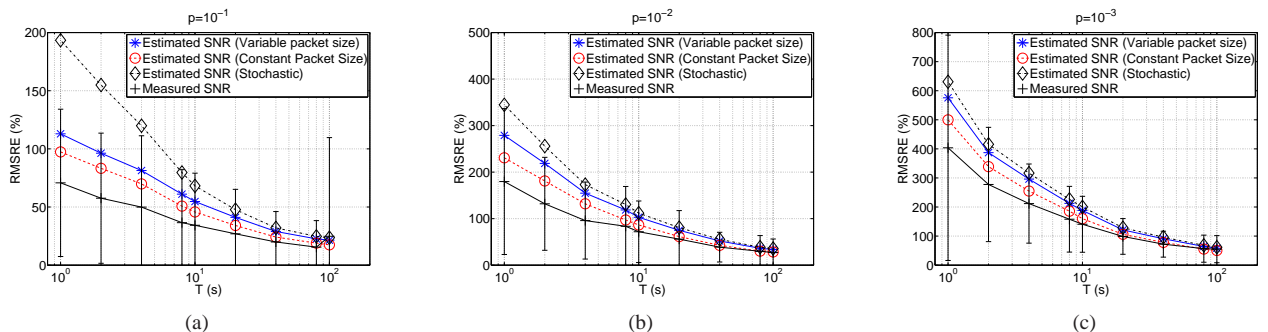


Fig. 13. RMSRE for sub-flows of trace 1 sampled at: (a)  $p = 0.1$ ; (b)  $p = 0.01$ ; (c)  $p = 0.001$ .

## VI. RELATED WORK

Many previous contributions have been made in the area of packet sampling, we review here those who are most relevant to our work.

The problem of estimating flow size distribution from sampled traces has been recently afforded in [22] and [10] using Maximum likelihood techniques. In particular, [10] focuses on the frequencies at which different numbers of packets per flow occur whereas [22] targets the flow size distribution tail index. The work [18] demonstrates that it is impossible in practice to recover the spectral density and the distribution of the number of packets per flow using traditional packet based sampling, even for high sampling rates. Moreover, it proposes to sample flows rather than packets in order to achieve higher accuracy at the expense of an increased computational complexity. In [28] and [26], the focus has been moved on flow size distribution of TCP flows. In particular, [28] exploits the theoretic formulation proposed in [26] to design a novel Dual

Sampling approach, combining the advantages of both packet and flow sampling.

In [9], a method is proposed to track the traffic flows through a domain by observing the trajectories of a subset of all packets traversing the domain, also in presence of unreliable transport of reports. The key idea is to sample packets with a hash function computed over the packet content. Using the same hash function in monitors yields the same sample set of packets in the entire domain, and enables the reconstruction of packet trajectories. In [5] a network wide optimal sampling platform has been proposed along a method to adaptively set the sampling rates of multiple measurements points in order to maximize the overall accuracy of the system.

FlexSample framework has been proposed in [25] to bias packet selection towards certain sub-populations of traffic, subject to an overall sampling constraint. The key idea behind FlexSample is that high-speed network devices can maintain approximate statistics using fast, space-efficient counters to determine the subpopulation to which each packet belongs;

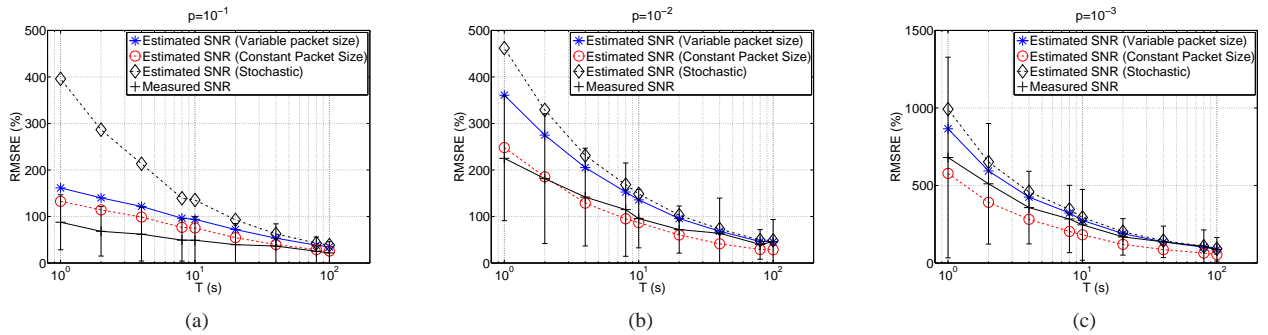


Fig. 14. RMSRE for sub-flows of trace 2 sampled at: (a)  $p = 0.1$ ; (b)  $p = 0.01$ ; (c)  $p = 0.001$ .

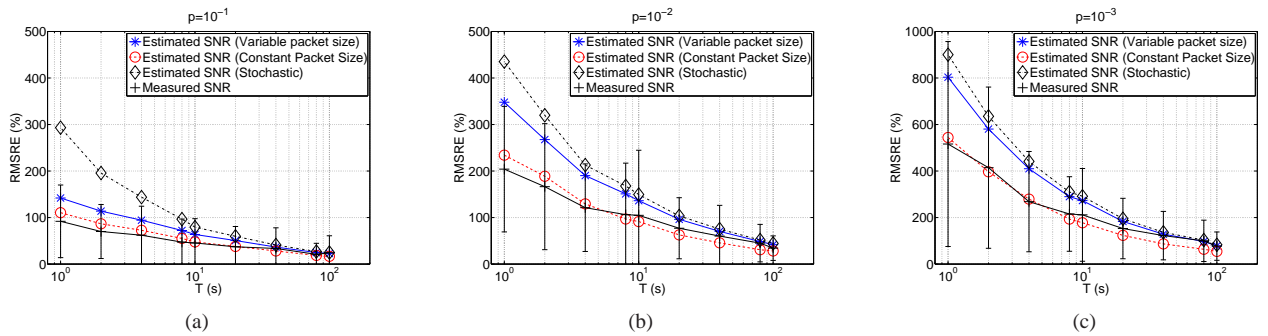


Fig. 15. RMSRE for sub-flows of trace 3 sampled at: (a)  $p = 0.1$ ; (b)  $p = 0.01$ ; (c)  $p = 0.001$ .

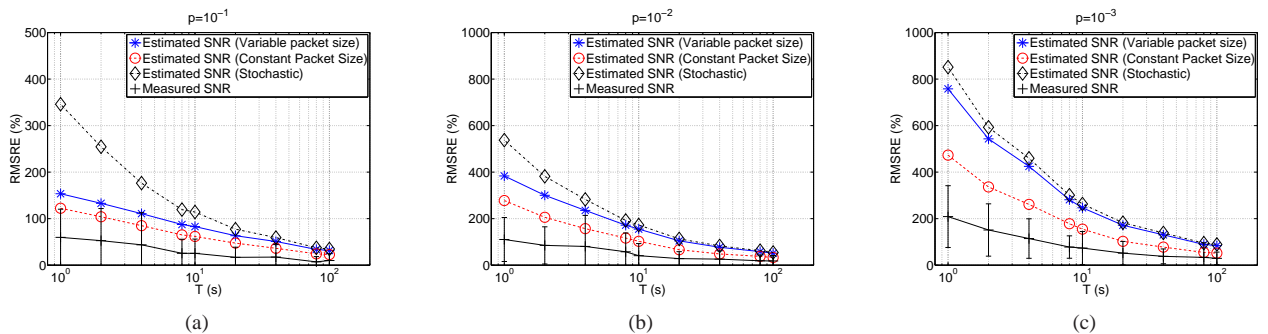


Fig. 16. RMSRE for sub-flows of trace 4 sampled at: (a)  $p = 0.1$ ; (b)  $p = 0.01$ ; (c)  $p = 0.001$ .

these counters can then be used to bias packet selection towards packets that belong to desired sub-populations.

In [11] a method for sampling flow records of a router is proposed. It is based on a threshold-based sampling strategy that sets the sampling probability according to the size of the flow records. The theoretical properties of the estimator have been derived. Moreover, it has been demonstrated that the algorithm has an accuracy slightly smaller than a modified version of the sample and hold algorithm proposed in [13]. Finally, several strategies to dynamically control the volume of the sampled traffic are proposed and compared.

In [21], the Sketch Guided Sampling (SGS) has been proposed as an alternative to uniform sampling. Using SGS, the packet sampling probability is set according to an estimate of the size of the flow to which the packet belongs. In this way, SGS is able to significantly increase the packet sampling rate of the small and medium flows at slight expense of the large flows, resulting in much more accurate estimations of various network statistics.

Finally, other interesting proposals have been conceived to deal with the detection and volume estimation of large flows [13], [23], [6], [16].

## VII. CONCLUSIONS

A novel technique to deal with packet sampling has been presented in this work. In particular, using a frequency domain analysis, useful SNR metrics have been derived to predict the impact of packet sampling on the estimation accuracy of the binned traffic spectrum. The proposed metrics, which have been validated using real traffic traces, consist of closed-form equations and, as a consequence, can be easily used for dimensioning today network monitoring and management systems. To provide a further insight, a comparison with an analogous expression derived using standard stochastic analysis has been provided, showing that, especially for low averaging time windows  $T$ , the frequency domain approach is able to outperform the accuracy of the stochastic one.

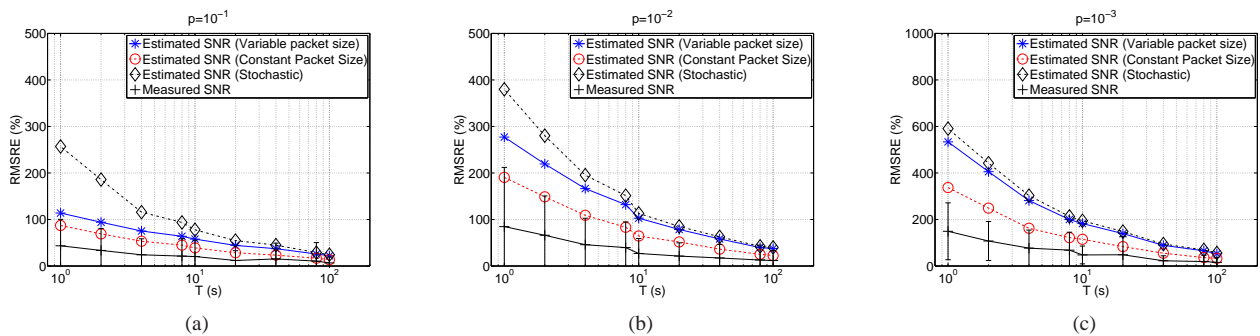


Fig. 17. RMSRE for sub-flows of trace 5 sampled at: (a)  $p = 0.1$ ; (b)  $p = 0.01$ ; (c)  $p = 0.001$ .

This ability to estimate original traffic rate at small time windows is very helpful for designing advanced monitoring applications that require high temporal resolutions. Current work focuses on the identification of such applications and on the extension of the presented approach into a network-wide setting where information coming from different routers are combined together for a better global estimation.

#### VIII. ACKNOWLEDGMENTS

Authors would kindly thank anonymous reviewers and Dr. Gennaro Boggia for the helpful advices they provide for improving the earlier versions of this work.

#### REFERENCES

- [1] Caida the cooperative association for internet data analysis. <http://www.caida.org/>.
- [2] Mawi working group traffic archive. <http://tracer.csl.sony.co.jp/mawi/>.
- [3] P. Barford, J. Kline, D. Plonka, and A. Ron. A signal analysis of network traffic anomalies. In *Proc. of ACM IMW 2002*.
- [4] D. Brauckhoff, B. Tellenbach, A. Wagner, M. May, and A. Lakhina. Impact of packet sampling on anomaly detection metrics. In *Proc. of ACM SIGCOMM IMC 2006*.
- [5] G. R. Cantieni, G. Iannaccone, C. Barakat, C. Diot, and P. Thiran. Reformulating the monitor placement problem: Optimal network-wide sampling. In *In Proc. of ACM CoNeXT 2006*.
- [6] B. Y. Choi, J. Park, and Z. L. Zhang. Adaptive packet sampling for accurate and scalable flow measurement. In *Proc. of IEEE Globecom 2004*.
- [7] K. C. Claffy, G. C. Polyzos., and K. W. Braun. Application of sampling methodologies to network traffic characterization. *ACM SIGCOMM Comput. Commun. Rev.*, 23(4), 1993.
- [8] N. Duffield. A framework for packet selection and reporting. In *IETF Draft (work in progress)*, Jun. 2008.
- [9] N. Duffield and M. Grossglauser. Trajectory sampling with unreliable reporting. *IEEE/ACM Trans. on Networking*, 16(1):37–50, 2008.
- [10] N. Duffield, C. Lund, and M. Thorup. Estimating flow distributions from sampled flow statistics. In *Proc. of ACM Sigcomm 2003*.
- [11] N. Duffield, C. Lund, and M. Thorup. Learn more, sample less: Control of volume and variance in network measurement. *IEEE Trans. on Information Theory*, 51(5):68–80, 2005.
- [12] C. Estan, K. Keys, D. Moore, and G. Varghese. Building a better netflow. In *Proc. of ACM SIGCOMM 2004*.
- [13] C. Estan and G. Varghese. New directions in traffic measurement and accounting: Focusing on the elephants, ignoring the mice. *ACM Trans. Comput. Syst.*, 21(3), 2003.
- [14] L. A. Grieco and C. Barakat. An analysis of packet sampling in the frequency domain. In *Proc. of ACM SIGCOMM IMC 2009*.
- [15] L. A. Grieco and C. Barakat. A frequency domain model to predict the estimation accuracy of packet sampling. In *Proc. of IEEE Infocom 2010*.
- [16] F. Hao, M. Kodialam, T. V. Lakshman, and S. Mohanty. Fast, memory efficient flow rate estimation using runs. *IEEE/ACM Trans. on Networking*, 15(6):1467–1477, 2007.
- [17] F. Harris. On the use of windows for harmonic analysis with the discrete fourier transform. *Proceedings of the IEEE*, 66(1), 1978.
- [18] N. Hohn and D. Veitch. Inverting sampled traffic. *IEEE/ACM Trans. on Networking*, 14(1):68–80, 2006.
- [19] P. Kanuparth, C. Dovrolis, and M. Ammar. Spectral probing, crosstalk and frequency multiplexing in internet paths. In *Proc. of ACM SIGCOMM IMC 2008*.
- [20] S. Katti, D. Katabi, C. Blake, E. Kohler, , and J. Strauss. Multiq: Automated detection of multiple bottlenecks along a path. In *Proc. of ACM IMC 2004*.
- [21] A. Kumar and J. Xu. Sketch guided sampling - using on-line estimates of flow size for adaptive data collection. In *Proc. of IEEE Infocom 2006*.
- [22] P. Loiseau, P. Goncalves, S. Girard, J. Kuntzmann, F. Forbes, J. Kuntzmann, and P. V.-B. Primet. Maximum likelihood estimation of the flow size distribution tail index from sampled packet data. In *Proc. of ACM SIGMETRICS 2009*.
- [23] T. Mori, M. Uchida, R. Kawahara, J. Pan, and S. Goto. Identifying elephant flows through periodically sampled packets. In *Proc. of ACM SIGCOMM IMC 2004*.
- [24] J. Proakis and D. G. Manolakis. *Digital Signal Processing*. Prentice Hall, Int. Eds., 3 edition, 1996.
- [25] A. Ramachandran, S. Seetharaman, N. Feamster, and V. Vazirani. Fast monitoring of traffic subpopulations. In *Proc. of ACM IMC 2008*.
- [26] B. Ribeiro, D. Towsley, T. Ye, and J. Bolot. Fisher information on sampled packets: an application to flow size estimation. In *Proc. of ACM IMC 2006*.
- [27] F. Silveira, C. Diot, N. Taft, and R. Govindan. Astute: detecting a different class of traffic anomalies. In *Proc. of ACM SIGCOMM 2010*.
- [28] P. Tune and D. Veitch. Fisher information on sampled packets: an application to flow size estimation. In *Proc. of ACM IMC 2008*.