Inverting Sampled Traffic

Nicolas Hohn
n.hohn@ee.mu.oz.au

Darryl Veitch
dveitch@unimelb.edu.au

Australian Research Council Special Research Center for Ultra-Broadband Information Networks
Department of Electrical and Electronic Engineering
University of Melbourne, Vic 3010, Australia

ABSTRACT

Routers have the ability to output statistics about packets and flows of packets that traverse them. Since however the generation of detailed traffic statistics does not scale well with link speed, increasingly routers and measurement boxes implement sampling strategies at the packet level. In this paper we study both theoretically and practically what information about the original traffic can be inferred when sampling, or 'thinning', is performed at the packet level. While basic packet level characteristics such as first order statistics can be fairly directly recovered, other aspects require more attention. We focus mainly on the spectral density, a second order statistic, and the distribution of the number of packets per flow, showing how both can be exactly recovered, in theory. We then show in detail why in practice this cannot be done using the traditional packet based sampling, even for high sampling rate. We introduce an alternative flow based thinning, where practical inversion is possible even at arbitrarily low sampling rate. We also investigate the theory and practice of fitting the parameters of a Poisson cluster process, modelling the full packet traffic, from sampled data.

Categories and Subject Descriptors

C.2.3 [**Computer-communications Networks**]: Network Operations – Network monitoring; G.3 [**Probability and Statistics**]:

General Terms

Measurement, Theory

Keywords

Thinning, sampling, traffic modeling, Internet data, long range dependence, TCP flows, transform inversion, Poisson cluster process

IMC'03, October 27–29, 2003, Miami Beach, Florida, USA.

1. INTRODUCTION

1.1 Motivation

Network traffic measurement is essential for traffic engineering (e.g. link upgrades or traffic re-routing) and traffic accounting (e.g. usage based pricing). Routers offer tools such as Cisco's Netflow [1] or Inmon's sFlow [2] that give information about the flows of packets that traverse them. However the generation of detailed traffic statistics does not scale well with link speed. This is why packet sampling techniques are increasingly being used in routers [3] to export the statistics of a portion of the traffic only. The problem that then immediately arises is how to deal with such partial measurements. One can think of this as a two step process: first recover the statistics of the full traffic from the retained sampled data through some inversion procedure, and second, take appropriate decisions based on the characteristics of the full traffic. While the second step is left to traffic engineers and managers, the first corresponds to an interesting and important task which has only recently been attracting attention. Our aim is to provide theoretical results for the problem of recovering statistics beyond first order from sampled traffic, and to see how successfully such results can be applied in practice with real traffic. We focus mainly on two statistics: the spectral density of the packet arrival process, and the distribution of the number of packets per flow. This implies that we limit ourselves to portions of traffic that can be considered stationary. It also means that we do not try to recover sample values, such as actual number of packets in flows on the measured link, but rather the distribution from which these samples were drawn.

Traffic statistics commonly considered vary widely depending on user requirements and the capabilities of the collection mechanism. In this paper we first place ourselves in a general framework whereby any raw statistics of the sampled data that we may need are considered to be available, as we focus primarily on the feasibility of the inversion problem. In some cases these statistics may not be readily available in today's routers, or they may be close to impossible to provide because of real-time constraints. For example few routers can export packet level statistics such as sizes and timestamps of individual packets. In addition, currently high-end routers use switched instead of shared backplanes, and therefore not all packets are seen at any single point of the backplane [4]. Purpose built link monitoring boxes however, or dedicated passive measurement infrastructures supporting offline studies based on sampled traffic, will be capable of much finer grained storage and processing.

1.2 Terminology

The definition of an Internet Protocol (IP) flow is central to this work. In the research community the generally accepted definition [5] is a set of packets with the same 5-tuple {IP protocol; source address; destination address; source port; destination port}, and with a fixed maximum inter-packet time T_0. On the other hand, IP flows are defined slightly differently in a router where a flow can be terminated due to: (i) timeout, but also (ii) protocol (FIN packet sent by the Transmission Control Protocol (TCP)) or (iii) memory management (the flow is terminated in order to free resources for new flows). Another definition worth mentioning is found in [6] where an adaptive timeout based on flow characteristics is used. In the rest of this paper we adopt the first definition, i.e. 5-tuple with static timeout. The actual value of the timeout T_0 will be discussed later.

In mathematical terms, we are mainly interested in the point process of packet arrival times and will not be concerned with packet sizes. In point process theory, the action of 'sampling' points along the real line is called *thinning*, and we will use these two terms interchangeably. From a theoretical perspective, one is interested in recovering as much information as possible about the original point process by observing a thinned version of it. We will use 'full' or 'original' to refer to the non-sampled packet traffic, and 'sampled' or 'thinned' to refer to the sampled traffic. The process of thinning the packet process is to be understood in general terms as the action of only recording part of the total traffic according to a certain rule.

In this paper we will study two different sampling rules: *packet thinning*, which acts directly on individual packets and is ignorant of flows, and *flow thinning*, where entire flows of packets are retained or discarded at once. Independent and identically distributed (i.i.d.) packet thinning consists of, for each packet in an independent manner, retaining the packet with probability q or discarding it with probability $1-q$. Similarly, i.i.d. flow thinning consists of, for each flow independently, leaving the flow untouched with probability q or removing it entirely with probability $1-q$.

We use a hierarchy of descriptors to study the statistics of packet traffic. We refer to *packet level* when describing statistics which do not use or refer to any imposed structure or detailed modelling assumptions. Examples of packet level statistics are the mean packet arrival rate or the spectral density of the packet arrival process. *Flow level* is concerned with statistics arising from the grouping of packets into flows, such as the distribution of the number of packets per flow. Finally, although of less importance here, we use *in-flow level* to refer to statistics describing the placement of packets within a flow, such as the mean arrival rate of packets belonging to a given flow.

1.3 Previous work

In the early 1990's, data collection on the T1 NSFNET backbone showed that information was lost during peak periods. Sampling methods were therefore advocated in [7] to reduce the load on the measurement infrastructure. The aim of this work was to estimate the packet size distribution from the sizes of sampled packets. Different sampling strategies were compared: deterministically taking one in every N packets (systematic sampling), taking on average one in N packets (simple random sampling) or taking one packet in every bucket of size N (stratified random sampling). In [8] an adaptive sampling rate was proposed to optimize the resource allocation. An adaptive sampling technique was also used in [9] where a bound on the sampling error for traffic load measurement was studied. Another study of sampling techniques can be found in [10] where the mean number of packets and the packet size distribution are estimated from a sampling where the number of skipped packets is a Poisson random variable. Sampling strategies were also used in [11] for the detection of denial of service attacks. In [12] estimates are provided from sampled traffic of the mean number of bytes or packets of a set of packets with common properties (e.g. protocol, IP addresses, Autonomous System,...). Because of the heavy tailed distribution of file sizes, a particular kind of sampling, known as stratified sampling [13], is used to reduce the variance of the estimators. It basically consists in sampling 'more' in the heavy tail of the distribution and gives different weights to different samples.

Each of the aforementioned studies were concerned with a packet level description of network traffic in the sense described above. Much closer in spirit to this paper is the work presented in [14], where it is shown how certain first-order IP flow level statistics can be recovered from sampled traffic. In particular, an estimator (and its all important variance) of the mean number of packets per flow is given. The estimation is not blind and makes strong use of additional information contained in the TCP packet header. More specifically the recovery scheme requires the knowledge of the number of original flows, and as it is assumed that this is not measured directly, it must be inferred separately. It is shown how this can be achieved by looking for TCP SYN packets in the case of 'ideal' TCP flows which all begin with a SYN packet and have infinite timeouts.

1.4 Outline and Main Contributions

We are interested in inverting sampled traffic in a statistical sense, focusing mainly on two quantities: the spectral density (packet level), and the distribution of the number of packets per flow (flow level). In section 2 we address this problem from a theoretical perspective. We first consider the case of *packet sampling* since it is the method currently implemented in routers. We show in particular how the theory of point processes can help recover the original spectral density from the thinned data. We also propose a theoretical scheme to recover the full distribution of the number of packets per flow. In this respect we extend the work of [14] where only the mean number of packets was recovered. We then present an alternative sampling technique named *flow sampling* which is as computationally feasible as packet sampling, but has a more straightforward inversion mechanism both at the packet and flow level. The inversion methods require different assumptions on the original traffic depending on the sampling method which are carefully detailled and justified.

The practical application of the two methods to real traffic and the limitations of their numerical evaluation are given in section 3. Our main contribution is the demonstration of the fact that inversion is essentially impossible in practice in the case of packet thinning for any useful thinning probability, whereas flow thinning can be usefully inverted no matter how high this probability becomes, provided enough traffic is sampled. Section 4 is concerned with the application of the sampling techniques to a recently introduced cluster point process model of backbone traffic. It is shown how

the parameters of the model can be fitted from the thinned data obtained from both sampling techniques in theory, not always in practice. Finally in section 5 we conclude and discuss the computational implications of the inversion techniques.

2. INVERTING SAMPLING: THEORY

In this section we study two different sampling techniques, which we call *i.i.d. packet sampling* and *i.i.d. flow sampling*. We present theoretical inversion methods to recover the spectral density and the distribution of the number of packets per flow from the observed thinned traffic. All quantities corresponding to thinned traffic will be written with the superscript $^{(q)}$, where q is the retention probability defined below.

2.1 Packet sampling

In general terms, the i.i.d. packet thinning of a stationary point process X with rate λ consists in independently keeping each point of X with probability q or rejecting it with probability $1-q$ to form a new point process $X^{(q)}$ with rate $\lambda^{(q)} = q\lambda$.

2.1.1 Packet level

The original rate can be recovered from that of the thinned process in a straightforward way via

$$\lambda = \frac{1}{q}\lambda^{(q)}. \tag{1}$$

A much less intuitive result links the spectral densities of X and $X^{(q)}$. From [15, 16], for any simple, locally finite and second order stationary point process X with spectral density $\Gamma_X(\omega)$, the spectral density of $X^{(q)}$ reads

$$\Gamma_X^{(q)}(\omega) = q^2\Gamma_X(\omega) + q(1-q)\lambda. \tag{2}$$

From equations (2) and (1) the spectrum $\Gamma_X(\omega)$ of the original process can therefore be recovered and reads

$$\Gamma_X(\omega) = \frac{1}{q^2}\Big(\Gamma_X^{(q)}(\omega) - (1-q)\lambda^{(q)}\Big). \tag{3}$$

This powerful result gives readily accessible and very useful information about the original process without making any assumptions on its detailled structure. In particular no modelling assumptions are required beyond stationarity.

2.1.2 Flow level

Let us assume that the original process is in fact the superposition of identically distributed groups of points called clusters. In the traffic context these are packets grouped into flows. Let P be the discrete random variable describing the number of points per cluster, with density $p_k = \Pr(P = k)$, distribution F_P, and finite mean $\mu_{\rm P}$. In practice no flow of length 0 is observed and therefore $p_0 = 0$. Let $P^{(q)}$ be the discrete random variable describing the number of packets per flow after packet thinning, with density $p_k^{(q)} = \Pr(P^{(q)} = k)$, and distribution $F_P^{(q)}$. In this subsection our aim is to recover the properties of the marginal F_P of the original flows from $F_P^{(q)}$. Since we look at the marginal only, there is no need to assume independence between flows.

Conditioning on the number of packets in a given original flow, the probability $p_k^{(q)}$ of having a flow of size $k \geq 0$ after thinning reads

$$\begin{aligned} p_k^{(q)} &= \sum_{j=k}^{\infty} \Pr\{\text{k packets after thinning} \\ &\qquad \mid \text{j packets before thinning}\} p_j \\ &= \sum_{j=k}^{\infty} \binom{j}{k} q^k (1-q)^{j-k} p_j. \end{aligned} \tag{4}$$

Equation (4) gives the densities of the thinned flows as a function of the densities of the original flows. To invert this relation we use results on probability generating functions and complex analysis.

Let us first introduce some notation. In the following $\mathcal{C}(z,r)$, $\mathcal{D}(z,r)$ and $\bar{\mathcal{D}}(z,r)$ will denote respectively the circle, the open disk and the full disk with center z and radius r. Denote by B the binomial random variable such that $\Pr(B=0) = 1-q$ and $\Pr(B=1) = q$. Let $G_{\rm P}(z)$, $G_{\rm P}^{(q)}(z)$ and $G_B(z)$ be the probability generating functions of P, $P^{(q)}$ and B defined respectively as

$$G_{\rm P}(z) = \sum_{j=0}^{\infty} p_j z^j\,, \quad G_{\rm P}^{(q)}(z) = \sum_{j=0}^{\infty} p_j^{(q)} z^j,$$

and $G_B(z) = 1 - q + qz$. $G_{\rm P}(z)$ and $G_{\rm P}^{(q)}(z)$ are defined on the closed unit disk $\bar{\mathcal{D}}(z,r) = \mathcal{D}(0,1) \cup \mathcal{C}(0,1)$, but if F_P is heavy tailed they are only analytic on the open unit disk $\mathcal{D}(0,1)$ due to a singularity at $z = 1$. $G_B(z)$ is an entire function (analytic for all $z \in \mathbb{C}$).

By definition of i.i.d. packet thinning, $P^{(q)}$ can be expressed as a sum of P i.i.d. binomial random variables. From results on the generating function of a compound distribution the following relation holds:

$$G_{\rm P}^{(q)}(z) = G_{\rm P}(G_B(z)) \quad \text{for} \quad z \in \bar{\mathcal{D}}(0,1). \tag{5}$$

This equation is the transform domain version of equation (4). Since $G_B^{-1}(\bar{\mathcal{D}}(0,1)) = \bar{\mathcal{D}}(1-q,q)$, one can obtain $G_{\rm P}$ from equation (5) as

$$G_{\rm P}(z) = G_{\rm P}^{(q)}\Big(\frac{z-(1-q)}{q}\Big) \quad \text{for} \quad z \in \bar{\mathcal{D}}(1-q,q). \tag{6}$$

Now, as we see from equation (5), the probabilities p_j that we wish to calculate can be obtained by picking out the coefficients of a power series expansion of $G_{\rm P}$ about the origin. However, equation (6) only gives an inversion formula for $G_{\rm P}$ for $z \in \bar{\mathcal{D}}(1-q,q)$, a closed disk which lies within the unit circle and is centered at $z_0 = 1-q$. (see the thick circle in figure 1(a)). It does not give us $G_{\rm P}$ over the full unit disk, nor an expansion about the origin. We consider how to circumvent these difficulties in a moment.

Using standard results on generating functions, the mean number of packets per flow can be recovered via

$$\mu_{\rm P} = \frac{dG_{\rm P}}{dz}\Big|_{z=1} = \frac{1}{q}\frac{dG_{\rm P}^{(q)}}{dz}\Big|_{z=1} = \frac{\mu_{\rm P}^{(q)}}{q}. \tag{7}$$

Let F_P be a heavy tailed distribution such that

$$1 - F_P(x) \overset{x\to+\infty}{\sim} \frac{L}{x^\alpha}, \tag{8}$$

where $L > 0$ and $1 < \alpha < 2$. From equations (8) and (6) one can show by using a Tauberian theorem [17, p.333] that

$F_P^{(q)}$ has tail behaviour

$$1 - F_P^{(q)}(x) \overset{x\to+\infty}{\sim} \frac{L^{(q)}}{x^\alpha}, \tag{9}$$

where

$$L^{(q)} = q^\alpha L. \tag{10}$$

The thinned distribution for the number of packets per flow is therefore also heavy tailed with the same index but reduced tail mass. In fact the Tauberian theorem used above is even stronger and gives an equivalence between equations (8) and (9). This means that if a heavy tailed is observed in the thinned traffic, it must come from the original traffic, and cannot have been created by the thinning process itself. From equation (10) one can trivially invert the tail prefactor:

$$L = \frac{1}{q^\alpha} L^{(q)}. \tag{11}$$

We now present two different theoretical schemes to recover the original probability densities.

Scheme 1: Analytic continuation

Our aim is to construct a power series expansion of $G_{\rm P}$ about the origin in order to recover the p_j via expansion on the left in equation (5). In principle, since $G_{\rm P}$ is analytic in $\mathcal{D}(0,1)$ and from equation (6) its values are known on $\mathcal{D}(1-q,q)$ which lies inside $\mathcal{D}(0,1)$, $G_{\rm P}$ is known on $\mathcal{D}(0,1)$ through *analytic continuation*. The required expansion about the origin can therefore be found. Carrying this through in practice however is not straightforward.

We denote by $z_0 = 1-q$ the origin of the original analytic domain $\mathcal{D}_0 = \mathcal{D}(z_0, q)$. Within $\mathcal{D}_0$ it is easy to show from equation (6) that the following power series expansion holds:

$$G_{\rm P}(z) = \sum_{n=0}^{\infty} a_n^0 (z - z_0)^n, \quad z \in \mathcal{D}_0. \tag{12}$$

where the coefficients obey

$$a_n^0 = \frac{p_n^{(q)}}{q^n} \tag{13}$$

and the radius of convergence is $r_0 = q$.

The basic principle we employ is to choose a point $z_1 \in \mathcal{D}_0$ and to expand $G_{\rm P}$ as a power series about it. The coefficients of this new series can then be obtained by comparing with the series of equation (12) evaluated at $z = z_1$, and are

$$a_j^1 = \sum_{n=j}^{\infty} \binom{n}{j} a_n^0 (z_1 - z_0)^{n-j}. \tag{14}$$

Consider how this works for the simple case of $q \in [0.5, 1]$ where we are able to choose z_1 to be the origin, as illustrated in figure 1(a) for $q = 0.6$. Substituting $a_n^0 = \frac{p_n^{(q)}}{q^n}$ into equation (14) and noting from equation (5) that $a_j^1 = p_j$ in this case, we have

$$p_j = \sum_{n=j}^{\infty} \binom{n}{j} \frac{(-1)^{n-j}}{q^n} (1-q)^{n-j} p_n^{(q)}, \tag{15}$$

which converges for $q \in [0.5, 1]$. An alternative way to derive this inversion formula is to directly apply a combinatorial identity to invert equation (4). The identity in question states ([18], p.49), with no convergence criteria given, that $B_k = \sum_{j=k}^{\infty} \binom{j}{k} A_j$ and $A_j = \sum_{k=j}^{\infty} \binom{k}{j} (-1)^{k+j} B_k$ are inverses. In the present context this identity can only help us for $q \in [0.5, 1]$, a very mild degree of thinning.

For $q \in [0, 0.5)$ z_1 cannot be chosen at the origin, and we adopt a recursive procedure involving a sequence $\{z_k\}$, $k = 1, 2, \cdots l$, of points along the real axis obeying $1 > z_0 > z_1 > \cdots z_l = 0$, z_l being the origin itself (figure 1(b) illustrates the case where $q = 0.1$ and $l = 5$). At the kth stage, z_k will be chosen to lie inside the circle of convergence $\mathcal{C}_{k-1}$ from the previous stage, and $G_{\rm P}$ will be expanded in a power series centered about z_k, whose coefficients a_j^k will be obtained through those of the previous stage:

$$a_j^k = \sum_{n=j}^{\infty} \binom{n}{j} a_n^{k-1} (z_k - z_{k-1})^{n-j}. \tag{16}$$

Since z_k lies inside the unit circle where we know $G_{\rm P}$ is analytic, its circle of convergence $\mathcal{C}_k$ will first encounter a singularity at $z = 1$, and so the corresponding radius of convergence will be $r_k = 1 - z_k$. In this way, as the sequence $\{z_k\}$ marches towards the origin the radii of convergence increase monotonically to 1. In fact the z_k can be chosen so that the origin is approached geometrically: a minimum of $\lceil -\log_2(q) \rceil$ iterations is required. As before, the coefficients of the final power series will be the desired densities, that is $p_j = a_j^l$.

Scheme 2: Cauchy integral

A second theoretical scheme to recover the original p_j is based on another important result of complex analysis: the *Cauchy integral formula*, which for our particular problem reads

$$p_j = \oint_{\mathcal{S}} \frac{G_P(z)}{z^{j+1}} dz, \tag{17}$$

where $\mathcal{S}$ can be any closed contour containing the origin. Inversion methods based on equation (17), including methods using inverse Fourier transforms and damping techniques, are summarized in [19]. They work well when one can directly evaluate $G_{\rm P}$ on a contour including the origin. (In some queuing problems for instance one has an explicit expression for the generating function to be inverted for the corresponding probability densities). Methods have also been developed to remove the aliasing terms caused by the unavoidable discretization of the integral in the numerical evaluation of equation (17), both when F_p is light tailed [20] and heavy tailed [21]. Note that for $q > 0.5$, we can choose $\mathcal{S} = \mathcal{C}_0$ and the Cauchy integral can be directly evaluated along this contour. However for $q < 0.5$ we first have to infer the values of $G_{\rm P}$ on some suitable contour $\mathcal{S}$ and then use equation (17) to recover the p_j.

A common method to do so is to use *Padé approximants*. It consists in approximating $G_{\rm P}$ at the point $z = z_0$ by a quotient of two polynomials $P(z)$ and $Q(z)$, of degree L and M respectively, and then evaluating $P(z)/Q(z)$ at the desired values of z. Details on the determination of these polynomials and convergence issues can be found for instance in [22]. In our case we evaluate the Padé approximants on a contour $\mathcal{S}$ chosen to be the unit circle. The main drawback of the Padé approximation is that there are no general bounds on the error. The natural bound in this case, that $|G_p(z)| \leq 1$ on the unit circle, is not of much use.

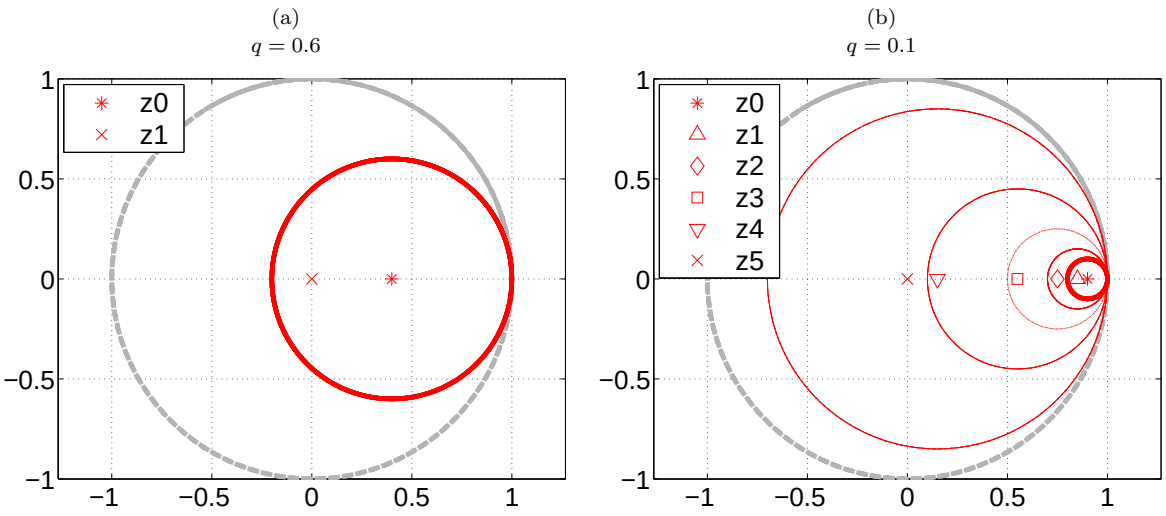

Figure 1: Analytic continuation method for (a) $q = 0.6$, and (b) $q = 0.1$. The thick solid dark circle represents $\mathcal{C}_0 = \mathcal{C}(z_0, q)$ and the thick dotted grey circle is the unit circle $\mathcal{C}(0, 1)$. For $q = 0.6$ z_1 can be chosen as the origin and an expansion made there whereas for $q = 0.1$ a series of analytic continuations are required, before a point, z_5, can be chosen as the origin.

While complex analysis provides elegant theoretical results for the recovery of F_{P} from $F_{\mathrm{P}}^{(q)}$, the procedures are quite involved. Moreover, it is an ill-posed problem in the sense that small errors in the evaluation of G_p at points in the original domain $\bar{\mathcal{D}}_0$ become magnified in the extrapolation [23]. To put this in perspective, in the case of significant thinning, say $q = 0.001$, we are trying to extrapolate values from a tiny circle of radius q close to $z = 1$ up to the entire unit circle. Note that equations (7) and (11) do not suffer from this problem as $z = 1$ is on the circle for all q. As we will see in section 3, the practical limitations of the two schemes described above are so severe that only a few values for the first iteration step can be obtained numerically. Given these fundamental difficulties at the flow level, we do not attempt to investigate the inversion of in-flow statistics for packet thinning. We now turn to a very different kind of sampling, *i.i.d. flow thinning*, which has quite different properties.

2.2 Flow sampling

As stated in the introduction, i.i.d. flow sampling consists in selecting flows with probability q. Flows will be taken to be identically distributed through an assumption of stationarity,

2.2.1 *Flow level*

Since the flows that are kept by the thinning procedure are identically the same as the original flows, all the marginal flow properties, and in particular the distribution P of the number of packets per flow, can be readily estimated from the observed thinned traffic. There is no inversion problem as such (beyond estimation issues), and the value of q plays no theoretical role. The same holds true for in-flow statistics, which are uncorrupted by flow thinning. This is in marked contrast with the packet thinning scenario and its problematic inversion requirements. As for packet thinning, no assumption of flow independence is needed at this point.

2.2.2 *Packet level*

We now explain how, under mild assumptions on the underlying process, packet level information such as the spectral density of X can be recovered.

Consider a special kind of stationary point process $X(t)$, well suited to flow thinning, known as a *Cluster Process*. Let the arrival times $\{t_{\mathrm{F}}(i)\}$ of flows follow a given process $Y(t)$ of rate λ_{F}. The cluster process $X(t)$ is defined as

$$X(t) = \sum_i \mathcal{G}_i(t - t_{\mathrm{F}}(i)), \tag{18}$$

where $\mathcal{G}_i(t)$ represents the arrival process of packets within flow i. It is assumed that the subsidiary process $\mathcal{G}_i(t)$ has a finite mean number μ_{P} of packets per flow and a finite intensity. These two conditions are necessary for $X(t)$ to be stationary [24]. Recent results on the spectral theory of point processes give the form of the spectrum of $X(t)$ in this general case [15]. However we will keep things simple in the following by assuming that $Y(t)$ is a Poisson process. This is justified by the fact that for IP traffic the true arrival process of flows has very little influence on the spectrum of the packet arrival process, at least for lightly loaded links [25, 26]. In addition we will now also take flows to be mutually independent. With these assumptions $X(t)$ becomes a *Poisson Cluster Process* (PCP).

Let $\Gamma_G(\omega)$ be the 'spectrum' of $\mathcal{G}_i(t)$, more precisely the expectation of the modulus squared of the Fourier transform of $\mathcal{G}_i(t)$. The spectrum of $X(t)$ can be shown to be simply [16]

$$\Gamma_X(\omega) = \lambda_{\mathrm{F}} \Gamma_G(\omega). \tag{19}$$

From [24] the rate of the stationary process $X(t)$ reads

$$\lambda = \lambda_{\mathrm{F}} \mu_{\mathrm{P}}. \tag{20}$$

Let us now consider the effect of flow thinning a PCP. We will make use of the well known independent splitting property of a Poisson process [16].

THEOREM 1. *Let $Y(t)$ be a Poisson process with rate λ. Independently classify each point either as type I with probability q, otherwise as type II. Let $Y_I(t)$ and $Y_{II}(t)$ denote the point processes composed of the type I or type II points. Then $Y_I(t)$ and $Y_{II}(t)$ are independent Poisson processes with rate λq and $\lambda(1-q)$ respectively.*

The i.i.d. sampling with probability q of the Poisson flow arrival process $Y(t)$ with rate λ_{F} is thus a Poisson process $Y^{(q)}(t)$ with rate $\lambda_{\text{F}}^{(q)} = q\lambda_{\text{F}}$. This means that flow sampling transforms $X(t)$ into a PCP $X^{(q)}(t)$ with flow rate $\lambda_{\text{F}}^{(q)}$ and the *same* $\mathcal{G}_i(t)$. The rate of $X^{(q)}(t)$ reads $\lambda^{(q)} = \lambda_{\text{F}}^{(q)}\mu_{\text{P}}^{(q)} = q\lambda_{\text{F}}\mu_{\text{P}} = q\lambda$ and the original rate can be recovered via

$$\lambda = \frac{1}{q}\lambda^{(p)}. \tag{21}$$

The density spectrum of $X^{(q)}(t)$ reads

$$\Gamma_X^{(q)}(\omega) = \lambda_{\text{F}}^{(q)}\Gamma_G(\omega) = q\Gamma_X(\omega), \tag{22}$$

from which the original density spectrum can be expressed as

$$\Gamma_X(\omega) = \frac{1}{q}\Gamma_X^{(q)}(\omega). \tag{23}$$

3. INVERTING SAMPLING: PRACTICE

The previous section was concerned with theoretical inversion methods for two different kinds of thinning. In this section we present a numerical evaluation of these inversion techniques. We begin with the packet level statistics in 3.1 before tackling the flow level statistics in 3.2. Results concerning the estimates of first order quantities and their confidence intervals for packet sampled traffic can be found in [14] and will not be detailed here.

The passive measurements used to illustrate the thinning methods are presented in table 1. They come from the Auckland-IV [27] and Abilene NLANR [28] trace repositories. The traffic can be considered stationary for the period of time covered by the traces.

Trace	Date	Local Time	Rate (Mbps)	Link
AUCK-d1	20010402	13:00 to 16:00	2.5	OC3
IPLS	20020814	10:00 to 10:10	418	OC48c

Table 1: Details of passive measurements.

3.1 Packet level

From equations (3) and (23), the spectrum of the full traffic can be recovered from the spectrum and the rate of the thinned traffic for both sampling techniques. When estimating from data however, because of the scaling properties of network traffic we use a wavelet based estimate of the spectral density. Because of the linearity of the relationship between the Fourier and wavelet spectra, essentially the same inversion formulae can be used. A full description of the wavelet approach can be found in [29] and we only briefly summarize it here.

The (discrete) wavelet transform of a process X is defined by coefficients $d_X(j,k) = \langle X, \psi_{j,k}\rangle$, where the family $\{\psi_{j,k}\}$ is derived from the mother wavelet $\psi(t)$, $j = \log_2(\text{scale})$, and $k \in \mathbf{N}$ indexes time at octave j. Let $X(t)$ be a continuous time stationary process with power spectral density $\Gamma_X(\nu)$. The variance of its wavelet coefficients satisfies:

$$\mathbb{E}|d_X(j,k)|^2 = \int \Gamma_X(\nu)2^j|\Psi(2^j\nu)|^2 d\nu, \tag{24}$$

where $\Psi(\nu)$ denotes the Fourier transform of ψ. If X possesses scale invariance over a range of scales, for example if it is long range dependent (LRD), defined as a power law divergence of the spectrum at the origin: $\Gamma_X(\nu) \sim c|\nu|^{-\alpha},\ |\nu| \to 0,$ with $\alpha \in (0,1)$, then in the limit of large scales equation (24) becomes

$$\mathbb{E}|d_X(j,k)|^2 \sim C2^{j\alpha},\ j \to +\infty. \tag{25}$$

Equation (24) can be viewed as defining a kind of wavelet energy spectrum well suited to the study of scaling processes. To estimate the wavelet spectrum from data, the simple time average based variance estimates: $\text{Var}(d_j) = \frac{1}{n_j}\sum_k |d_X(j,k)|^2$, where n_j is the number of $d_X(j,k)$ available at scale j, perform very well, because of the short range dependence in the wavelet domain [29]. A plot of the logarithm of these estimates against j we call the *Logscale Diagram* (LD), in which straight lines indicate scaling.

LDs are in fact closely related to traditional Fourier spectrum. As a rough approximation, one would recover the (log of the) spectral density as a function of frequency simply by reversing the scale axis in the LD plot. Because equation (24) is linear in $\Gamma_X(\nu)$, and an energy preserving wavelet normalization is being used, the relations linking the spectrum of the thinned and full traffic are also valid for the wavelet spectrum.

Figure 2 illustrates the inversion methods in the (log) wavelet domain. The thick gray line corresponds to the wavelet spectrum of the original traffic, while the vertical lines mark confidence intervals on the spectrum estimate at the different scales. The straight line observed over large scales betrays long memory.

When q is relatively large ($q = 0.1$), the spectrum inferred from the packet thinned traffic is remarkably close to the 'true' spectrum estimated directly from the full traffic, as illustrated on figure 2(a). The fact that fine details of the spectrum can be reproduced is due to the fact that equation (2) is valid for any second order stationary point process. On the other hand, the spectrum reconstructed from the flow thinned traffic does not match the true spectrum quite as well. While very good at large scales, the reconstruction fails to precisely match the small scale behaviour. This is a direct consequence of our assumption underlying the inversion formula that flows are uncorrelated. The inversion is incapable of re-inserting the flow dependencies which were weakened by the thinning. Despite this strong assumption however, the inverted spectrum clearly reproduces the main features of the true spectrum.

When one moves to much smaller values of q however, as figure 2(b) shows for $q = 0.001$, the flow based thinning still gives a qualitatively accurate estimate while the inversion technique based on packet thinning is highly inaccurate. In fact, from the form of equation (3) one can see that the original spectrum $\Gamma_X(\omega)$ is recovered by measuring the difference between $\Gamma_X^{(q)}(\omega)$ and a Poisson noise. Since for small q the confidence intervals on the estimation become so large that $\Gamma_X^{(q)}(\omega)$ cannot be reliably distinguished from this noise, the inversion procedure must fail. The problem clearly becomes steadily worse as q drops. This is significant since as

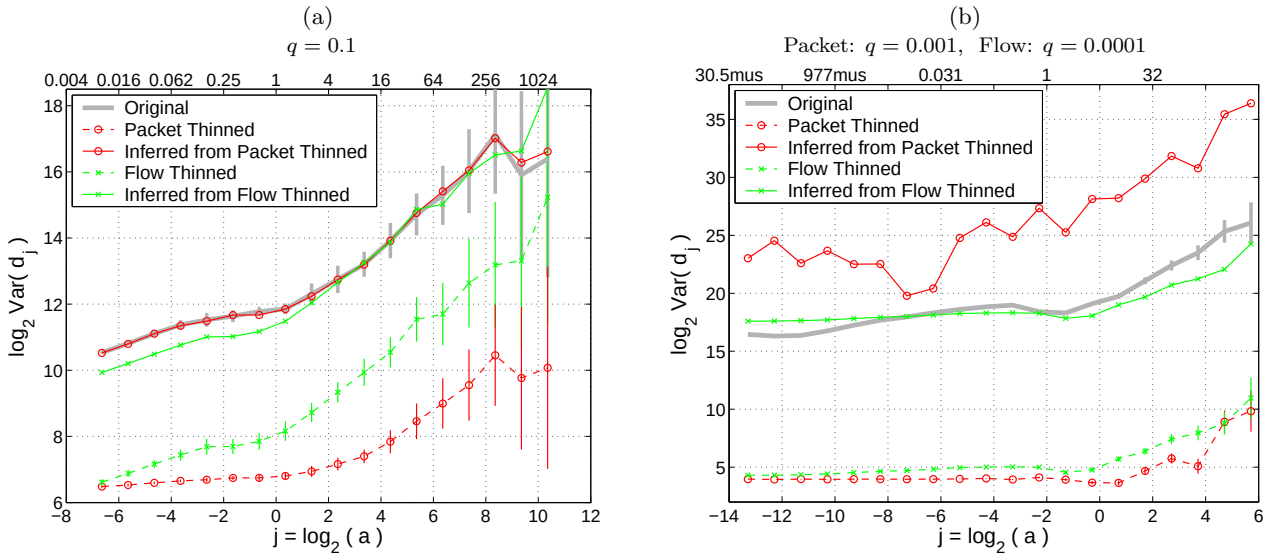

Figure 2: Spectrum reconstruction: (a) AUCK-d1: Logscale diagrams of the original traffic, packet and flow thinned traffic each with $q = 0.1$, and the two corresponding inverted estimates for the full traffic ($T_0 = 64$s). The top axis marks the timescale in seconds. (b) IPLS: Logscale diagrams of the original traffic, packet thinned traffic with $q = 0.001$, flow thinned traffic with $q = 0.0001$, and the two corresponding inverted estimates for the full traffic. Despite the fact that the flow thinned traffic is ten times thinner, the estimate recovered from it is far better.

link rates increase a trend to ever more aggressive thinning seems likely.

In contrast to the above, recovery of the spectrum in the case of flow thinning does not suffer from the same drawback as it simply involves multiplying $\Gamma_X^{(q)}(\omega)$ by a scale factor (an upward translation on the logarithmic scale of the LD). In fact, the quality of the estimation through the flow thinning inversion method depends mainly on the number of flows N remaining after thinning, the value of q being largely irrelevant. At constant N, the inversion method will therefore lead to an approximately 'constant' error, irrespective of the thinning probability. This point is illustrated in figure 3 where inversion based estimates are given for two values of q at constant N. In practice however, non-stationarities and edge effects make it difficult to accurately estimate the spectrum when the number of remaining flows drops too low (In the case of the traffic used in figure 3, although there were 3 million flows, the trace was only 10 minutes long resulting in quite strong edge effects). The near independence of the inversion method with respect to q is a strong argument in favour of flow based sampling for spectral estimation.

3.2 Flow level

In general, estimating the distribution of the number of packets per flow consists of an estimation of the densities $p_j^{(q)}$ for the thinned process, followed by an inversion phase. For flow thinning we have already seen that the inversion is trivial, and the $p_j^{(q)}$ can be estimated from a histogram. For packet thinning however even the first phase is potentially problematic, as a knowledge of $p_0^{(q)} > 0$ is needed. Since the proportion of discarded flows is not automatically observed as it is in flow thinning, the quantity $p_0^{(q)}$ cannot be estimated without extra information.

The simplest solution is to supply the total number N_F of flows with the measured sampled traffic, in the spirit of Inmon's sFlow [2]. Another solution proposed by [14] was already mentioned in the introduction. Assume that each original TCP flow has only one packet with a SYN flag and that it is the first. Consider the set of such SYN packets. It is clear that the probability that a given SYN packet is retained is also q, and that this is therefore the probability that a flow has been retained. Let $N_F^{(q)}$ be the total number of observed flows, and $N_1^{(q)}$ the number of observed packets with a SYN flag. An estimate of the number of flows N_F before thinning is $N_F = N_1^{(q)}/q$. One can construct an estimate of $p_0^{(q)}$ via $p_0^{(q)} = N_F^{(q)}/N_F$.

Another important practical issue with packet thinning concerns the consistency of the flow definition before and after thinning. For example, in order to prevent the breakup of flows due to their sparsity after thinning, one should at least replace the timeout value T_0 with T_0/q. However this does not eliminate all problems and extra flows can still be created for some types of applications [14]. It is another advantage of flow thinning that problems of this type do not arise. The flow definition and timeout value adopted for the full traffic applies without change after sampling.

To clearly evaluate the performance of the thinning inversion techniques in isolation from other issues such as those above, we assume in what follows that $p_0^{(q)}$ is known. In addition, we will first assume that we know the distribution of P and can therefore evaluate $p_k^{(q)}$ numerically from equation (4). For this purpose we use a simple discrete Pareto-like variable H with distribution

$$F_{\mathrm{H}}(k; a, \beta) = 1 - (ak + 1)^{-\beta} \sim 1 - Lk^{-\beta}, \quad k = 1, 2, \cdots, \tag{26}$$

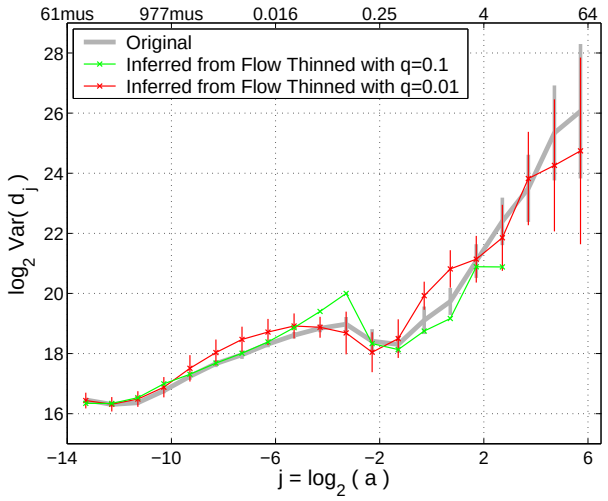

Figure 3: Reconstruction of the (log wavelet) spectral density from flow thinning when the number of flows after thinning is constant (N=3000). The quality of the estimation remains roughly unchanged as q varies. The confidence intervals for $q = 0.1$ (omitted for clarity) are similar to those of $q = 0.01$.

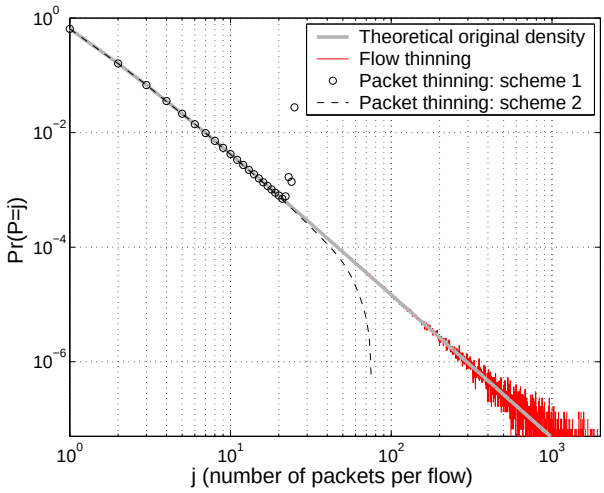

Figure 4: Inversion of the p_j, light thinning: Numerical evaluation of the different inversion schemes for $q = 0.6$ using F_P given by equation (26) with $a = 1$ and $\beta = 1.5$. Packet thinning inversion: Scheme 1: even with no estimation, the inversion becomes unstable for small j. Scheme 2: Some improvement at high computational cost. ($L = M = 200$ and 2^{15} discretization steps for the evaluation of the Cauchy integral). Flow thinning inversion and estimation: starting with 10^6 flows, estimates are reliable, extend to much greater j, and degrade gracefully.

where $a = L^{-1/\beta} > 0$ is a scale parameter. The mean of H is $\mathbb{E}[H] = a^{-\beta}\zeta(\beta, 1/a)$ for $\beta > 1$, and is therefore fully determined by the tail behaviour. The variance is infinite.

We first consider inversion scheme 1 (using analytic continuation) in the case where $q > 0.5$ for which we can calculate p_j from equation (15), which we repeat here:

$$p_j = \sum_{n=j}^{\infty} \binom{n}{j} \frac{(-1)^{n-j}}{q^n} (1-q)^{n-j} p_n^{(q)}. \qquad (27)$$

There are three main issues with the numerical evaluation of this sum. First, it must be truncated at some $n = n_{max}$. Second, before the asymptotic decay of the terms takes over, their magnitudes become enormous due to the q^{-n} factor. The alternating sign cancels these out, but as the sum, being a probability, must lie in $[0, 1]$ the precision necessary is very large unless q is close to 1. Finally, in practice there are the additional errors from the need to estimate the $p_n^{(q)}$.

Numerical results for scheme 1 are presented in figure 4 where the truncation issue has been carefully addressed, the estimation issue does not apply as exact values are used, but where nonetheless precision limitations creates serious problems. They show that using typical double precision (`Matlab` was used), only the first 20 or so values of p_j can be recovered from equation (27) with $q = 0.6$ before dramatic numerical instability sets in. The numerical evaluation of scheme 2 (Padé approximants followed by the Cauchy integral formula) takes us a little further, but at the price of a fairly intensive numerical evaluation. It was found that increasing the degree of the Padé approximants did not significantly improve the accuracy of the calculations. In contrast to these packet thinning based inversion schemes, the 'inversion' from flow thinning, including the numerical estimation of the $p_n^{(q)}$, provided a low cost and reliable estimation of p_j, whose accuracy dropped gracefully as j increased as seen in figure 4. The estimates of the $p_n^{(q)}$ were made according to the following formula, where $p_0^{(q)}$ was assumed known:

$$p_j^{(q)} = (1 - p_0^{(q)}) o_j^{(q)} \quad \text{for} \quad j \geq 1, \qquad (28)$$

where $o_1^{(q)}, o_2^{(q)} \ldots$ are the normalized histogram estimates of the number of packets per flow after flow thinning.

In the general case where $q \in (0, 1]$, algorithms to recursively compute the coefficients $a_j^{(k)}$ based on the analytic continuation idea of scheme 1 can be found in [30] and [31]. However since the particular form of the coefficients (equation (13)) prevents a precise numerical evaluation of even the first step of the recursion, the method is not applicable here. When the thinning procedure removes more than half of the original packets, the inversion method for packet thinning cannot be used in practice unless 'infinite precision' arithmetic is employed. This is unlikely to be computationally feasible in a router context.

Again, flow based thinning avoids the above problems, and just as for the spectrum, the quality of the estimation depends essentially on the number of flows N after thinning, and not on the value of q. This is shown in figure 5 where the complementary cumulative distribution function (CCDF) of the number of packets per flow on an OC-48 link, and the estimated CCDFs for three different values of q, are plotted at constant $N = 3000$. As expected, the quality of the estimation of the CCDF is roughly independent of q. However, since the estimation of heavy tails is a notoriously difficult problem [32], one should make sure that N will be large enough to allow a sufficiently precise estimation of the distribution tail.

In summary, despite the fact that both sampling types can theoretically be inverted, the numerical study carried

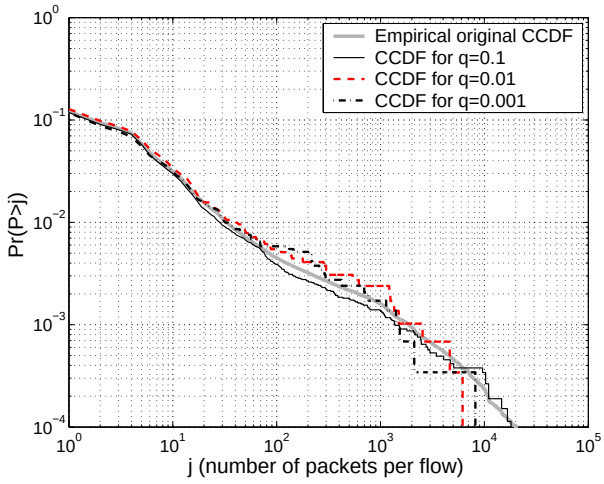

Figure 5: Inversion of the p_j, heavy thinning: Empirical CCDF of the number of packets per flow for the IPLS trace, and estimated CCDFs obtained from flow thinned traffic for q $\{0.1, 0.01, 0.001\}$ while the number of flows after thinning remains constant ($N = 3000$). The quality of the estimation remains unchanged despite the wide variation in q.

out in this section reveals the following:

- The *packet sampling* technique leads to an excellent reconstruction of the spectrum and a fair estimate of the p_j for j up to the order of 50 for $q > 0.5$. However in the useful range $q \ll 0.5$, the quality of the spectrum estimate is poor and deteriorates steadily as q gets smaller and it becomes impossible to evaluate the p_j even for small j as soon as q drops below 0.5 without extended precision arithmetic.

- The *flow sampling* technique gives a reasonable estimate of the spectrum and an excellent estimate of the p_j for a large range of thinning probabilities. In particular, for thinning probabilities used in practice, of the order of 1% or less, flow thinning is by far superior to packet thinning if one is interested in recovering detailed characteristics of the original traffic such as its spectrum or the distribution of flow size.

It is worth noting at this point that if a parametric family was chosen for F_P one could try to estimate its parameters, a much easier task than trying to numerically recover each p_j. Since no family has been identified as valid for all Internet flows, we have not pursued this here.

4. THE BARTLETT-LEWIS PROCESS

In this section we apply both packet and flow thinning to a particular kind of Poisson Cluster Process which is well suited for modelling lightly loaded but highly aggregated links such as those in network backbones. We present the point process model and its main properties in section 4.1, before deriving the thinning results in section 4.2. The viability of fitting the model via measurements on thinned data is investigated in section 4.3.

4.1 Definition

It was shown in [33] that a particular case of a PCP known in the literature as a Bartlett-Lewis point process (BLPP) [16] is well suited to model Internet backbone traffic. It is a particular case of a PCP (see the definition in section 2.2 and equation (18)) where each cluster $\mathcal{G}_i(t)$ is a *finite renewal process*. In such a case $\mathcal{G}_i(t)$ reads

$$\mathcal{G}_i(t) = \sum_{j=1}^{P(i)} \delta\Big(t - \sum_{l=1}^{j-1} A_i(l)\Big), \tag{29}$$

where $\delta(t)$ is a delta function centered at $t = 0$, $A_i(l)$ denotes the $l-$th inter-arrival for flow i, and the inner sum is defined to be zero if $j = 1$. $P(i)$ is the number of packets in flow i, *normalised so that* $p_0 = 0$. In what follows we will assume that the packet inter-arrival times within flows, described by the random variable A, are Gamma distributed with characteristic function

$$\Phi_A(\omega) = (1 - ib\omega)^{-c} \tag{30}$$

and density $f(x)$, where $c > 0$ is the shape parameter and b is the scale parameter. The mean and standard deviation are given by $\mu_A = bc$ and $\sigma_A = b\sqrt{c}$. The rate of a cluster is given by $\lambda_A = 1/\mu_A$. Further discussion on the role and meaning of the model parameters can be found in [33].

The benefits of the BLPP are numerous: not only is the BLPP a model with physically meaningful parameters and inherently positive marginals, but also there exists analytical expressions for many of its statistics. The spectrum of the BLPP is of particular interest here. Expressions for it can be found for instance in [16, p.315] and [34, p.79], and can be written in the form:

$$\Gamma_X(\nu) \;=\; \lambda_{\mathrm{F}}\Big(\frac{\mu_{\mathrm{P}}}{\lambda_{\mathrm{A}}}\Gamma_{\mathcal{G}}(\nu) + \big(S_{\mathcal{G}}(\omega) + S_{\mathcal{G}}(-\omega)\big)\Big), \tag{31}$$

where $\Gamma_{\mathcal{G}}(\nu)$ is the spectral density of the stationary renewal process with the same parameters as the finite flow renewal process, namely

$$\Gamma_{\mathcal{G}}(\nu) = \lambda_{\mathrm{A}}\Big[(1 - \Phi_A(\omega))^{-1} + (1 - \Phi_A(-\omega))^{-1} - 1\Big], \tag{32}$$

and

$$\mathrm{Re}(\, S_{\mathcal{G}}(\omega)\,) = \frac{\Phi_A(\omega)}{(1 - \Phi_A(\omega))^2}\Big(G_{\mathrm{P}}(\Phi_A(\omega)) - 1\Big). \tag{33}$$

As expected equation (31) is consistent with the general form for the spectral density of a PCP given in equation (19).

4.2 Thinning Bartlett Lewis point processes

If the PCP model is to be useful in practice, for example for the dimensioning of backbone links, one needs to be able to measure its parameters from data. It is therefore of interest to see if it is compatible with either or both of the thinning procedures. In this subsection we derive the properties of thinned BLPPs.

THEOREM 2. *An i.i.d. packet thinned Bartlett-Lewis process $X^{(q)}$ is also a Bartlett-Lewis process with parameters:*

- *flow rate:* $\lambda_F^{(q)} = \lambda_F(1 - p_0^{(q)})$,

- *density of* $P^{(q)}$*:* $x_j^{(q)} = \frac{p_j^{(q)}}{1 - p_0^{(q)}}$, $j > 0$, *and* $x_0^{(q)} = 0$,

- *density of in-flow packet inter-arrivals:*

$$f^{(q)}(x) \quad = \quad \mathcal{L}^{-1}\left[\frac{q\tilde{f}(s)}{1-(1-q)\tilde{f}(s)}\right].$$

PROOF. See Appendix. □

This agreable closure property of a BLPP, which is worth mentioning in its own right, also helps to make the inversion of its parameters analytically tractable.

THEOREM 3. *An i.i.d. flow thinned Bartlett-Lewis process $X^{(q)}$ is also a Bartlett-Lewis process with flow rate $\lambda_F^{(q)} = q\lambda_F$, $x_j^{(q)} = p_j$, and $f^{(q)}(x) = f(x)$.*

PROOF. The result follows from the discussion at the end of section 2.2. □

We see that the BLPP model has almost ideal theoretical properties with respect to the interpretation of thinned forms of itself, and the parameter inversion problem. In the next section we briefly consider the practical side of the question.

4.3 Fitting from thinned data

With respect to i.i.d. packet thinning, despite the attractive theoretical properties described above, most of theorem 2 cannot be exploited in practice if $q > 0.5$. The reasons are the same as those stated in section 3.2 concerning the recovery of the p_j from the $x_j^{(q)}$. Moreover, even if one could numerically evaluate these, there would be another inversion problem to recover the in-flow packet inter-arrival density from its Laplace transform, with similar limitations due, ultimately, to the very small values of all the $x_j^{(q)}$ except at $j = 1$ if q is small. For completeness, we note that the relevant inversion techniques are also based on Cauchy's integral formula and are similar to the one presented in section 2.1.2. They can be implemented using the Fast Fourier Transform [19].

We turn then to fitting from i.i.d. flow thinned data, where the simple inversion of theorem 3 presents no difficulties. Figure 6 illustrates the procedure for $p = 0.001$. The remarkable thing about this approach is that we do not need to explicitly invert the more complex in-flow or even flow level statistics. One merely fits the model on the thinned data as one would normally, and then scales up the value of λ_F. Figure 6 shows that the results can be good even for $p = 0.001$, and as before, it is to an excellent approximation only the total number of flows which determines the size of the confidence intervals, not q.

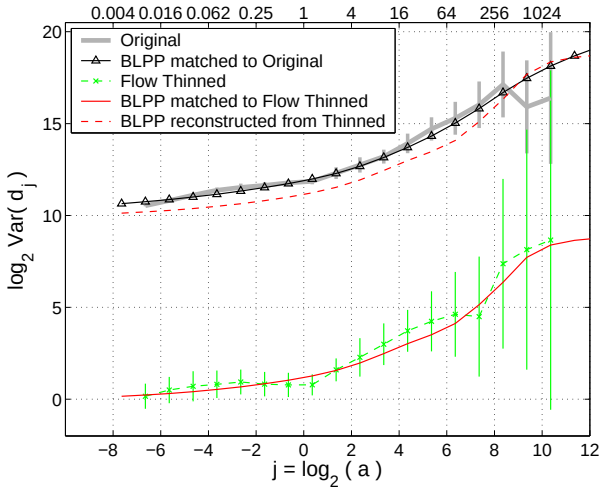

Figure 6: BLPP parameter fitting from flow thinned traffic AUCK-d1 data, thinned with $p = 0.001$, is matched to the BLPP, the theoretical spectrum calculated, and then inverted by simply shifting it vertically. The inversion compares well with the original data showing that the model can be successfully fitted from thinned data. The same fitting procedure applied to the full traffic is also shown.

5. CONCLUSIONS

We have explored in detail the question of recovering the spectrum and the distribution of the number of packets per flow of the packet arrival process, from sampled data. Two kinds of sampling were used, i.i.d. packet sampling, and i.i.d. flow sampling, with a given probability q of retaining a packet or flow respectively. In each case, exact theoretical inversion techniques were derived. However, in the case of packet thinning, we showed how the inversion methods were of little to no use in practice for q small enough to be truly useful, such as $q = 0.01$ or smaller, and become much worse as q becomes smaller still. An exception to this is the asymptotic tail which can be recovered by a different technique (although in practice it remains a very difficult problem). In sharp contrast, as flow thinning preserves flows intact but simply reduces their number, it avoids these problems entirely and the inversion is trivial. The performance of inversion methods based on flow thinning does not deteriorate with q but depends essentially on the number of retained flows, which could be set in practice depending on memory and computational limitations. However, the inversion step does assume flow independence, and so cannot capture all aspects of the traffic, whereas packet thinning based methods can provided q is large enough. However, for backbone links where there is strong evidence that dependence between flows is weak, this may not be important.

In practice, any attempt to gather flow statistics involves classifying individual packets into flows. For packet thinning, the process is simple since only the retained packets have to be classified. On the other hand, for flow thinning, all packets have to be put into flows before they can be discarded. This involves more computation and more memory if one uses the traditional hash table approach with one entry per flow. However this might not be such a drawback if new flow classification techniques, such as bitmap algorithms [35], can be applied instead. The total number of packets stored, for a given q, is essentially the same for both types of thinning.

We also investigated the fitting of a useful type of cluster model describing packet arrivals. It was shown that the model class is closed under both kinds of thinning and that exact inversion is theoretically possible. In practice however, again inversion based on packet thinned packet is not feasible for realistic values of q, whereas inversion based on fitting from flow based thinning performs well.

Acknowledgement

This work was funded by the Australian Research Council.

6. REFERENCES

[1] Cisco Netflow, `http://www.cisco.com/warp/public/732/Tech/netflow`.

[2] Inmon Corporation, *sFlow accuracy and billing*, `http://www.inmon.com/PDF/sFlowBilling.pdf`.

[3] Cisco Sampled NetFlow, `http://www.cisco.com/en/US/products/sw/iosswrel/ps1829/products_feature%_guide09186a0080081201.html#wp1019824`.

[4] G. Iannaccone, C. Diot, I. Graham, and N. Mc Keown, "Monitoring very high speed links," in *Proc. ACM/SIGCOMM Internet Measurement Workshop*, 2001.

[5] K.C. Claffy, H.-W. Braun, and G.C. Polyzos., "Parameterizable methodology for Internet traffic flow profiling," *IEEE Journal on Selected Areas in Communications*, vol. 136, no. 8, pp. 1481–1494, 1995.

[6] B. Ryu, D. Cheney, and H. Braun, "Internet flow characterization: Adaptive timeout strategy and statistical modeling," in *Proc. Passive and Active Measurement workshop*, 2001.

[7] K.C. Claffy, G.C. Polyzos, and H.-W. Braun., "Application of sampling methodologies to network traffic characterization," in *Proc. ACM SIGCOMM*, 1993, pp. 13–17.

[8] J. Drobisz and K.J. Christensen, "Adaptive sampling methods to determine network traffic statistics including the Hurst parameter," in *Proc. IEEE Annual Conference on Local Computer Networks*, 1998, pp. 238–247.

[9] B.-Y. Choi, J. Park, and Z.-L. Zhang, "Adaptive random sampling for total load estimation," in *Proc. IEEE International Conference on Communications*, 2003, pp. 1552–1556.

[10] G. Cheng and J. Gong, "Traffic behavior analysis with Poisson sampling on high-speed network," in *Proc. ICII 2001*, 2001, pp. 158–163.

[11] Y. Huang and J. M. Pullen, "Countering denial-of-service attacks using congestion triggered packet sampling and filtering," in *Proc. International Conference on Computer Communications and Networks*, 2001, pp. 490–494.

[12] N. Duffield, C. Lund, and M. Thorup, "Learn more, sample less: control of volume and variance in network measurement," *submitted*, 2003.

[13] W. Cochran, *Sampling Techniques*, Wiley, 1987.

[14] N.Duffield, C. Lund, and M. Thorup, "Properties and Prediction of Flow Statistics from Sampled Packet Streams," in *Proc. ACM/SIGCOMM Internet Measurment Workshop*, 2002.

[15] P. Bremaud, L. Massoullié, and A. Ridolfi, "Power spectra of random spike fields and related processes," *(submitted)*, 2003.

[16] D. J. Daley and D. Vere-Jones, *Introduction to the Theory of Point Processes*, Springer-Verlag, New York, 2nd edition, 2002.

[17] N.H. Bingham, C.M. Goldie, and J.L. Teugels, *Regular Variation*, Cambridge University Press, Cambridge England, 1987.

[18] J. Riordan, *Combinatorial Identities*, Wiley and Sons, 1968.

[19] J. Abate and W. Whitt, "The Fourier-series method for inverting transforms of probability distributions.," *Queueing Systems*, vol. 10, pp. 5–88, 1992.

[20] J. Daigle, "Queue length distributions from probability generating functions via Fourier transforms," *Operations Research Letters*, , no. 8, pp. 229–236, 1989.

[21] M. Roughan, D. Veitch, and M. Rumsewicz, "Computing queue length distributions for power-law queues," in *Proc. INFOCOM*, 1998, pp. 356–363.

[22] H. Amindavar and J. Ritchey, "Padé approximations of probability density functions," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 30, no. 2, pp. 416–424, 1994.

[23] K. Miller, "Stabilized numerical analytic prolongation with poles," *SIAM Journal on Applied Mathematics*, vol. 183, no. 2, pp. 346–363, 1970.

[24] P. A. W. Lewis, "A branching Poisson process model for the analysis of computer failures," *Journal of the Royal Statistical Society*, vol. 26, no. 3, pp. 398–456, 1964.

[25] N. Hohn, D. Veitch, and P. Abry, "The impact of the flow arrival process in Internet traffic," in *Proc. IEEE ICASSP*, 2003, pp. VI 37–40.

[26] N. Hohn, D. Veitch, and P. Abry, "Does fractal scaling at the IP level depend on TCP flow arrival processes ?," in *ACM/SIGCOMM Internet Measurement Workshop*, Marseille, France, 2002, pp. 63–68.

[27] Waikato Applied Network Dynamics, `http://wand.cs.waikato.ac.nz/wand/wits/`.

[28] National Laboratory for Applied Network Research, `http://www.nlanr.net/`.

[29] P. Abry, P. Flandrin, M. S. Taqqu, and D. Veitch, "Wavelets for the analysis, estimation, and synthesis of scaling data," in *Self-Similar Network Traffic and Performance Evaluation*, K. Park and W. Willinger, Eds., pp. 39–88. Wiley, 2000.

[30] P. Henrici, *Applied and Computational Complex Analysis, Vol 1*, Wiley and Sons, 1974.

[31] C. Chaffy, "The analytic continuation process: from computer algebra to numerical analysis," in *Proc. ACM International symposium on symbolic and algebraic computation*, 1994, pp. 216–222.

[32] R. J. Adler, R. E. Feldmann, and M. S. Taqqu, *A practical guide to heavy tails*, Birkhäuser, 1998.

[33] N. Hohn, D. Veitch, and P. Abry, "Cluster Processes, a Natural Langage for Network Traffic," *IEEE Transactions on Signal Processing, Special Issue on Signal Processing in Networking*, vol. 51, no. 8, pp. 2229–2244, 2003.

[34] D.R. Cox and Valerie Isham, *Point Processes*, Chapman & Hall, 1980.

[35] C. Estan, G. Varghese, and M. Fisk, "Bitmap algorithms for counting active flows on high speed links," in *Proc. ACM SIGCOMM Internet Measurement Conference*, 2003.

Appendix
Proof of Theorem 2, section 4.1

Let X be a BLPP and $X^{(q)}$ the process resulting from its i.i.d. packet thinning with probability q. The thinned flows are clearly i.i.d. with marginal distribution $F_{\mathrm{P}}^{(q)}$ given by equation (4). Since $p_0^{(q)} = \sum_{j=1}^{\infty}(1-q)^j p_j > 0$, in this picture λ_{F} is unchanged but some flows may be empty. To conform to a convention where a BLPP has zero probability of an empty flow, we must renormalise the $p_j^{(q)}$ from equation (4) to obtain a $F_{\mathrm{P}}^{(q)}$ with densities $x_j^{(q)} = \frac{p_j^{(q)}}{1-p_0^{(q)}}$, $j > 0$, and $x_0^{(q)} = 0$. The average flow arrival rate is then reduced to $\lambda_F^{(q)} = \lambda_F(1 - p_0^{(q)})$.

It is known [16] that if X is a renewal process with inter-arrival density $f(x)$, then the i.i.d. thinned process $X^{(q)}$ is another renewal process, with inter-arrival density $f_q(x)$ whose Laplace transform $\tilde{f}_q(s)$ reads

$$\tilde{f}_q(s) = \frac{q\tilde{f}(s)}{1-(1-q)\tilde{f}(s)}. \tag{34}$$

It follows that each finite ordinary renewal process that constitutes a flow of X will become another ordinary renewal process with the inter-arrival density above provided it has at least 2 points.

The remaining property of $X^{(q)}$ to specify is the arrival process of the non-empty thinned flows. We now show that this is in fact a Poisson process with rate $\lambda_F^{(q)}$. Since the flow evaporation probability $p_0^{(q)}$ acts independently on flows, by theorem 1 on Poisson splitting the original flow starting points (which may have themselves been thinned) of flows which do not evaporate form a Poisson process O of rate $\lambda_F^{(q)}$. Consider such a flow which has survived thinning. There exists a random variable $T \geq 0$ giving the time interval between the original starting point and the first non-thinned point after thinning in that flow. As this can be viewed as an i.i.d. translation by T of the points of O, which by a well known theorem [16] is another Poisson process of the same rate, the result follows.