Sampling vs Sketching: An Information Theoretic Comparison

Paul Tune and Darryl Veitch
Department of Electrical and Electronic Engineering
University of Melbourne, Australia
Email: {lsptune@ee., dveitch@}unimelb.edu.au

***Abstract*—The main approaches to high speed measurement in routers are traffic sampling, and sketching. However, it is not known which paradigm is inherently better at extracting information from traffic streams. We tackle this problem for the first time using Fisher information as a means of comparison, in the context of flow size distribution measurement. We first provide a side-by-side information theoretic comparison, and then with added resource constraints according to simple models of router implementations. Finally, we evaluate the performance of both methods on actual traffic traces.**

I. Introduction

The importance of fine grained router measurements has increased significantly in recent years, and this is set to continue. These escalating ambitions collide with the high speed of core Internet routers where both time and memory are scarce resources, limiting the breadth, depth, and accuracy of measurements and their reporting. Two approaches address this difficulty: *sampling* and *sketching*.

In sampling a subset of the incoming traffic is selected. It is the dominant approach currently, for example it is used in Cisco's *Netflow* [1], which is based on a simple pseudo-random per-packet sampling scheme. Sustainable sampling rates are commonly quoted to be 1 in 256 packets. The sampling literature has focused on countering the significant estimation problems arising from such a low sampling rate and simple sampling scheme (e.g. [5], [10], [19]).

Sketching refers to a set of techniques, originating from the database literature on streaming data models [2], [22], whereby compact data structures with fast update rules were used to approximately measure properties over the stream. Here the literature has focused on developing variants of existing methods capable of being implemented in the networking context, improving their accuracy and resource usage (e.g. [3] and the Space-Code Bloom filter [15] for approximate set and multiset membership testing respectively), and designing new sketching approaches to access a broader set of traffic metrics (e.g. entropy [32], heavy hitters [6], subpopulation flow size distributions [13]).

Although there have been a number of works combining sampling and sketching in some form [14], [17], [26], the traffic sampling and sketching worlds have remained largely separate. Sketching papers have referred to sampling simply to point to their implementation costs in order to motivate their work, whereas sampling papers ignore sketching altogether. Consequently, there is a dearth of detailed comparisons.

The goal of this work is to build insights which have broad validity, to determine and compare the *inherent* ability of sampling and sketching to extract information from the traffic stream, rather than to compare the performance of particular methods and the estimators built upon them. We argue that clearer understanding of their underlying comparative capabilities is essential if hybrid schemes are to be systematically designed, rather than invented ad hoc as they are currently. To that end, inspired by recent work [25], [28], [29], we perform an information theoretic comparison based on Fisher information.

The traffic metric we focus on is the *flow size distribution*. By *flow* we mean a set of packets that share a common key that can be readily computed over IP packet header fields. This metric is challenging as well as important for security, traffic engineering, and management. For example a SYN flood attack could be detected by observing an increase in the number of flows of size 1, and it also contains the 'heavy hitter' (rate elephant) flows.

We perform the comparison in the context of particular sampling and sketching schemes, namely *flow sampling* (FS) studied by Hohn et al. [10] among others, and the *counter array* approach of Kumar et al. [12]. We select FS as it has been recently shown to outperform other sampling alternatives with respect to Fisher information [28], [29]. We select the counter array sketch as it is built directly on a simple form of a canonical component of sketching methods: hash-based indexing (with collisions). We compare first from a pure information sense, and then with constraints corresponding to simple implementation models, to allow a comparison of achievable performance. Our main finding is that the two methods are surprisingly close, in terms of information theoretic potential. However, when constraints on memory and cost were included, while FS is generally superior, the sketch may surpass flow sampling under certain conditions.

The paper is organized as follows. Section II defines the methods we study in detail and describes our Fisher information based framework allowing them to be compared. We then derive the Fisher information matrix and its inverse explicitly for each method. Section III compares the methods theoretically, and then provides simple implementation models allowing a meaningful comparison given real-world constraints. Numerical evaluations are provided to illustrate and extend the analytic results. Section IV applies the methods

to real Internet traces. We conclude in Section V.

Note: in this extended and corrected version some proofs (which were omitted for space reasons in the Infocom proceedings [**?**]) are now provided in the Appendix. Corrections to the Infocom version and related additions (mainly in Section IV) are marked in red text.

II. Information Analysis

In this section we introduce the conceptual framework and provide required background on Fisher information. We then define the chosen methods: FS and counter sketch, and derive their key information theoretic properties, thereby extending to the latter the information treatment of FS developed in [28]. Resource constraints and implementation aspects are ignored here.

A. Traffic model and Fisher Information

Consider a measurement interval of duration T containing N_f flows. We assume that N_f is known (see Section III-B). Of these flows, M_k have size k packets, $1 \le k \le W$, where W is the maximum flow size. There are precisely $n = \sum_{k=1}^{W} kM_k$ packets in total and the average flow size is $D = n/N_f$. This is a deterministic model of the data over the measurement interval; randomness will enter in later only through the action of the measurement method itself.

Let $[m]$ denote the index set $\{1, 2, \ldots, m\}$ and let $\theta_k = M_k/N_f$. The flow size distribution, the unknown vector parameter we seek information on, is $\boldsymbol{\theta} = [\theta_1, \theta_2, \ldots, \theta_W]^T$, and obeys

$$0 < \theta_k < 1 \ , \ k \in [W], \quad \sum_{k=1}^{W} \theta_k = 1. \tag{1}$$

We write vectors in bold lower-case, matrices in bold upper case, $\mathbf{A}^{\mathrm{T}}$ denotes the transpose of $\mathbf{A}$, and $\mathbf{diag}(\mathbf{x})$ a $m \times m$ diagonal matrix whose diagonal entries are taken from the vector $\mathbf{x} \in \mathbb{R}^m$.

The traffic summary methods define the observable, which is a packet count C. This is a random variable, taking integer values $j \ge 0$, whose probability distribution $c_j(\boldsymbol{\theta})$ depends on the details of the method as well as $\boldsymbol{\theta}$. Viewed as a function of $\boldsymbol{\theta}$ for a fixed value j of the observed data, it is known as the *likelihood*: $f(j, \boldsymbol{\theta}) = c_j(\boldsymbol{\theta})$.

The *Fisher information* is a well known measure of the amount of information the observable holds about the unknown parameters:

$$\begin{aligned} \mathbf{J}(\boldsymbol{\theta}) &= \mathbb{E}[(\nabla_{\boldsymbol{\theta}} \log f(j;\boldsymbol{\theta}))(\nabla_{\boldsymbol{\theta}} \log f(j;\boldsymbol{\theta})^{\mathrm{T}}] \\ &= \sum_{j \ge 0} (\nabla_{\boldsymbol{\theta}} \log f(j;\boldsymbol{\theta}))(\nabla_{\boldsymbol{\theta}} \log f(j;\boldsymbol{\theta}))^{\mathrm{T}} c_j. \end{aligned} \tag{2}$$

Recall that an $n \times n$ real symmetric matrix $\mathbf{A}$ is *positive (semi)definite* if for all vectors $\mathbf{z} \in \mathbb{R}^n \backslash \{\mathbf{0}\}$, $(\mathbf{z}^{\mathrm{T}}\mathbf{A}\mathbf{z} \ge 0)$ $\mathbf{z}^{\mathrm{T}}\mathbf{A}\mathbf{z} > 0$. We write $\mathbf{A} \ge 0$ when $\mathbf{A}$ is positive semidefinite. The Fisher information obeys $\mathbf{J} \ge 0$, but not $\mathbf{J} > 0$ in general.

The great importance of the Fisher information lies in its connection to estimation variance. It is known that, if it exists, the *Cramér–Rao Lower Bound* (CRLB) $\mathbf{J}^{-1}$ is the lower bound on the covariance matrix $\boldsymbol{\Sigma}_{\boldsymbol{\theta}}$ of any unbiased estimator of $\boldsymbol{\theta}$, i.e. $\boldsymbol{\Sigma}_{\boldsymbol{\theta}} \ge \mathbf{J}^{-1}$ in the positive semidefinite sense.

The above definition of $\mathbf{J}$ is in the familiar, unconstrained case. Here, however, $\boldsymbol{\theta}$ is subject to the constraints in (1). Constraints provide additional knowledge of $\boldsymbol{\theta}$ without a single measurement being taken, and results in higher $\mathbf{J}$, and hence lower covariance. Only equality constraints actually result in information gain [7]. The constrained CRLB is [7]

$$\boldsymbol{\mathcal{I}}^{+} = \mathbf{J}^{-1} - \mathbf{J}^{-1}\mathbf{G}\left(\mathbf{G}^{\mathrm{T}}\mathbf{J}^{-1}\mathbf{G}\right)^{-1}\mathbf{G}^{\mathrm{T}}\mathbf{J}^{-1} \tag{3}$$

where $\boldsymbol{\mathcal{I}}^{+}$ denotes the Moore-Penrose pseudo-inverse [9, Chapter 20, pp. 493-514] of the constrained Fisher information matrix $\boldsymbol{\mathcal{I}}$. Here the *gradient* matrix is $\mathbf{G}(\boldsymbol{\theta}) = \nabla_{\boldsymbol{\theta}}(\mathbf{1}_W^{\mathrm{T}}\boldsymbol{\theta} - 1) = \mathbf{1}_W$, where $\mathbf{1}_W$ is a $W \times 1$ vector of ones. With a single constraint, $\boldsymbol{\mathcal{I}}^{+}$ has rank $W - 1$ and so is singular.

Of great importance to us are the diagonal entries of $\boldsymbol{\mathcal{I}}^{+}$, since $\mathrm{Var}(\hat{\theta}_k) \ge (\boldsymbol{\mathcal{I}}^{+})_{kk}$ for any estimator. Comparison of these between methods corresponds to comparing the best performance the schemes are capable of supporting, thereby reflecting their comparative efficiency in extracting information from the traffic stream.

B. Flow Sampling

Conceptually, FS is very simple: flows are sampled independently with probability p_f and dropped with probability q_f. Sampling a given flow means that each packet within it is sampled, otherwise none of its packets are. Here the variable C represents the random size of a *typical sampled flow*, once the measurement interval is over.

We follow the approach of [28], where the importance of assuming N_f to be both known and large is pointed out for sampling strategies in general. The key consequence is that we are not restricted to observations which are conditioned on at least one packet being sampled (the traditional understanding). Instead, for a typical flow randomly selected from the N_f available, $j = 0$ sampled packets is also a valid observation. The resulting *unconditional* distribution for the sampled flow size then takes the simple form

$$c_j = \sum_{k=1}^{W} b_{jk}\theta_k, \quad 0 \le j \le W, \tag{4}$$

or in matrix notation $\mathbf{c} = \mathbf{B}\boldsymbol{\theta}$ where $\mathbf{c} = [c_0, c_1, c_2, \ldots, c_W]^{\mathrm{T}}$. Here b_{jk} is simply the probability that if the original flow had k packets, only j remain after sampling. The *sampling matrix* $\mathbf{B}$ characterizes the sampling method, and is column stochastic.

The simplicity (and linearity) of the unconditional likelihood enables $\mathbf{J}$ and hence $\boldsymbol{\mathcal{I}}^{+}$ to be obtained explicitly [28]:

$$\mathbf{B} = \begin{bmatrix} q_f \mathbf{1}_W^{\mathrm{T}} \\ p_f \, \mathbf{diag}(\mathbf{1}_W) \end{bmatrix}, \tag{5}$$

$$\mathbf{J} = p_f \, \mathbf{diag}(\theta_1^{-1}, \theta_2^{-1}, \ldots, \theta_W^{-1}) + q_f \mathbf{1}_W \mathbf{1}_W^{\mathrm{T}}, \tag{6}$$

It follows that $\boldsymbol{\mathcal{I}}^{+} = \mathbf{J}^{-1} - \boldsymbol{\theta}\boldsymbol{\theta}^{\mathrm{T}} = \frac{1}{p_f}\left(\mathbf{diag}(\boldsymbol{\theta}) - \boldsymbol{\theta}\boldsymbol{\theta}^{\mathrm{T}}\right)$, and

$$(\boldsymbol{\mathcal{I}}^{+})_{kk} = \frac{\theta_k(1 - \theta_k)}{p_f} \text{ for all } k \in [W]. \tag{7}$$

C. Counter Sketch

The counter sketch introduced by Kumar et al. [12] consists of an array of A packet counters, initialized to zero at the beginning of the measurement interval. Incoming packets are mapped independently and uniformly over the counters, using a hash function mapping a flow key, so that each packet in a flow maps to the same counter, but collisions occur so that a given counter may sum the packet counts from two or more flows. We define $\alpha = N_f/A$ to be the *flow load factor*, the average number of flows per counter. Here C is the final packet count in a *typical counter*, at the end of the measurement interval when all N_f flows are in the sketch.

For space reasons, proofs in this section have been sketched only.

The Likelihood $f(j, \boldsymbol{\theta}) = c_j(\boldsymbol{\theta})$. From the i.i.d. nature of flow insertion, it is not hard to see that the counter value can be expressed as a sum of independent random variables $\{kX_k\}$:

$$C = \sum_{k=1}^{W} kX_k, \tag{8}$$

where X_k, which is binomially distributed with parameter $(M_k, 1/A)$, gives the number of flows of size k in the counter, each of which contributes k packets to C. However, C itself is far from binomial. When computing the distribution of C, as j increases, a combinatorial explosion ensues. To deal with it we turn to generating functions, where the independence of each flow 'type' (size) leads to the product $C^*(s) := \sum_{j=0}^{n} c_j s^j = \prod_{k=1}^{W}(1 - 1/A + s^k/A)^{M_k}$, $|s| \le 1$.

The last expression was simplified by approximating the distribution of X_k as Poisson with parameter $\lambda_k = M_k/A = \alpha\theta_k$, yielding

$$C^*(s) = \prod_{k=1}^{W} e^{\lambda_k(s^k-1)} = e^{-\alpha} e^{\sum_{k=1}^{W} \lambda_k s^k}. \tag{9}$$

This is justified if we assume that N_f is large with non–vanishing load α, which is both realistic, and in any case necessary in order to approximate the different counters as independent, whereas in fact they are dependent since their sum over the sketch must equal n.

Using (9) we can show the following, which is all we need to derive the Fisher information.

Theorem 1: For $j > 0$ and $1 \le i \le W$, the distribution of C is given by the recursion relation

$$\frac{\partial c_j}{\partial \theta_i} = \begin{cases} \alpha c_{j-i} & \text{if } j \ge i, \\ 0 & \text{otherwise.} \end{cases}$$

Proof: For $j = 0$, $c_0 = \lim_{N_f\to\infty}(1 - 1/A)^{N_f} = e^{-\alpha}$ since $N_f = A\alpha$. Using (9)

$$\frac{\partial}{\partial \theta_i} C^*(s) = \alpha s^i e^{-\alpha} e^{\sum_{j=1}^{W} \lambda_j s^j} = \alpha s^i C^*(s)$$

since by the properties of generating functions, the term s^i "right shifts" the index i of each term of $C^*(s) = \sum_{j=0}^{n} c_j s^j$. The result follows by equating coefficients. ∎

The first few probabilities are given by

$$c_0 = e^{-\alpha}, \quad c_1 = \lambda_1 e^{-\alpha},$$
$$c_2 = \left(\lambda_2 + \frac{\lambda_1^2}{2!}\right) e^{-\alpha}, \; c_3 = \left(\lambda_3 + \lambda_2\lambda_1 + \frac{\lambda_1^3}{3!}\right) e^{-\alpha},$$
$$c_4 = \left(\lambda_4 + \lambda_3\lambda_1 + \frac{\lambda_2^2}{2!} + \lambda_2\frac{\lambda_1^2}{2!} + \frac{\lambda_1^4}{4!}\right) e^{-\alpha}.$$

We denote the probability density vector of C by $\mathbf{c} = [c_0, c_1, c_2, \ldots]^{\mathrm{T}}$.

Unconstrained Fisher Information $\mathbf{J}$. Using Theorem 1, we have $\frac{\partial}{\partial\theta_i}\log f(j;\boldsymbol{\theta}) = \frac{1}{c_j}\frac{\partial c_j}{\partial\theta_i} = \frac{\alpha c_{j-i}}{c_j}$, where $\frac{\partial c_j}{\partial\theta_i} = 0$ for $i > j$, since it is impossible for flows larger than j to have contributed to a counter value of j. Let $\mathbf{C} = \mathbf{diag}(c_1^{-1}, c_2^{-1}, \ldots)$. It follows that

$$\mathbf{J} = \alpha^2 \mathbf{H}^{\mathrm{T}}\mathbf{C}\mathbf{H}, \tag{10}$$

where

$$\mathbf{H} = \begin{bmatrix} c_0 & 0 & 0 & \ldots \\ c_1 & c_0 & 0 & \ldots \\ c_2 & c_1 & c_0 & \ldots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} \begin{bmatrix} \mathbf{I}_W \\ \mathbf{0}_{W\times W} \\ \mathbf{0}_{W\times W} \\ \vdots \end{bmatrix} \equiv \check{\mathbf{H}} \cdot \mathbf{E}, \tag{11}$$

where $\mathbf{0}_{W\times W}$ is the $W \times W$ null matrix. Here $\mathbf{H}$ is just the first W columns of the matrix $\check{\mathbf{H}}$, which is a *Toeplitz operator* with generating sequence $\{c_j\}_{j\ge 0}$. The matrix $\mathbf{J}$ is symmetric and positive definite.

Constrained Fisher Information $\boldsymbol{\mathcal{I}}^+$. The inverse of $\mathbf{J}$ is

$$\mathbf{J}^{-1} = \alpha^{-2}\mathbf{H}^{+}\mathbf{C}^{-1}(\mathbf{H}^{+})^{\mathrm{T}}. \tag{12}$$

Since $\check{\mathbf{H}}$ is Toeplitz, so is its inverse $\check{\mathbf{H}}^{-1}$, with a generating sequence simply related to that above.

By using this and the properties of $\mathbf{H}$, equation (3) can be evaluated to

$$\boldsymbol{\mathcal{I}}^+ = \frac{1}{\alpha^2}\mathbf{H}^{+}\left[\mathbf{C}^{-1} - \frac{\bar{\mathbf{c}}_W \bar{\mathbf{c}}_W^{\mathrm{T}}}{\sum_{k=1}^{W} c_k \Big(\sum_{j=0}^{W-1} c'_j\Big)^2}\right](\mathbf{H}^{+})^{\mathrm{T}}, \tag{13}$$

where

$$\bar{\mathbf{c}}_W = \left[\Big(\sum_{k=0}^{W-1} c'_k\Big)c_1, \Big(\sum_{k=0}^{W-2} c'_k\Big)c_2, \ldots, c'_0 c_W, 0, \ldots\right]^{\mathrm{T}},$$

and $\{c'_k\}$ is a sequence formally related (corresponding to a 'negative' Poisson rate) to $\mathbf{c}$, which can be readily calculated.

Finally, the desired explicit expression for the diagonals is

$$(\boldsymbol{\mathcal{I}}^+)_{kk} = \frac{1}{\alpha^2}\left(\sum_{j=1}^{k} c'^2_{k-j} c_j - \frac{(\sum_{j=1}^{k}(\sum_{\ell=0}^{W-j} c'_\ell)c'_{k-j}c_j)^2}{\sum_{i=1}^{W} c_i \Big(\sum_{j=0}^{W-1} c'_j\Big)^2}\right). \tag{14}$$

Bounds on $\boldsymbol{\mathcal{I}}^+$. The following upper bound, from [7, Remark 4], formalizes the intuition that constraints add information.

Theorem 2: $\boldsymbol{\mathcal{I}}^+ \le \mathbf{J}^{-1}$.

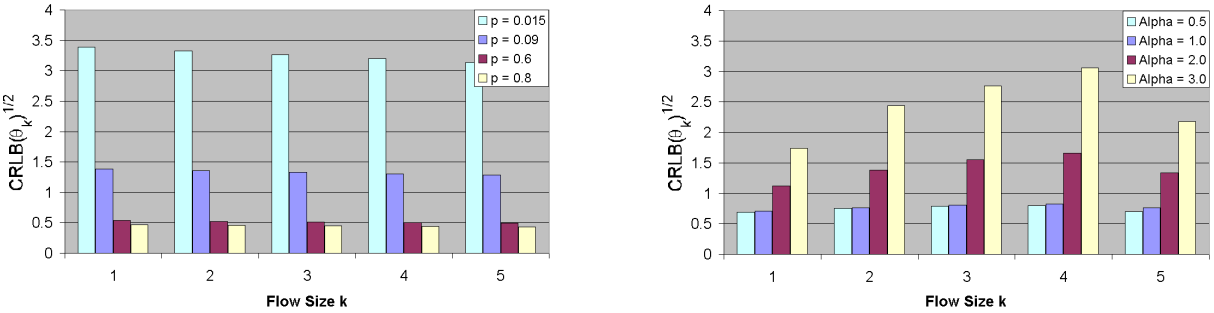

Fig. 1. CRLB for a truncated exponential with $W = 5$, $D \approx 2.9$: Left: flow sampling, Right: counter sketch.

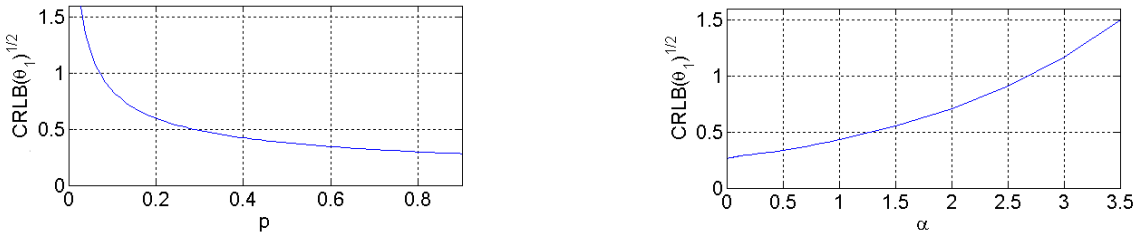

Fig. 2. CRLB for θ_1 for a truncated exponential with $W = 20$, $D \approx 8.9$: Left: p_f (flow sampling), Right: α (counter sketch).

By evaluating the exact expressions for a number of cases, we found this bound to be tight for $\alpha < 3$, a practical useful range.

We now derive a lower bound by providing the counter sketch with more information, specifically, the number of flows in each counter. This leads to a new sketching method, of interest in its own right, explored elsewhere. Intuitively, providing more information leads to a lower CRLB. By comparing the counter array sketch to the new method, we obtain the following.

Theorem 3: For all $k \in [W]$, $(\boldsymbol{\mathcal{I}}^+)_{kk} \geq \frac{e^\alpha}{\alpha}\theta_k(1-\theta_k)$.

Proof: We first derive the Fisher information matrix using the 2D generating function of the joint distribution of (C, F), where F is the Poisson(α) distributed flow counter variable, and then derive the corresponding $\boldsymbol{\mathcal{I}}^+$ using a similar approach to that above. The diagonal entries of the new method are $\frac{e^\alpha}{\alpha}\theta_k(1-\theta_k)$, for $k \in [W]$. By the data processing inequality [31], the Fisher information of the new method dominates that of the canonical sketch in the positive semidefinite sense, and hence in terms of diagonal elements as a special case. ■

III. Comparison

In FS flows are distinct but can be lost, whereas in the counter sketch none are lost but their identity becomes confused due to flow collision. The nature of their ambiguity is therefore entirely different. In sampling in general, it relates to flow sizes *above* the sampled size, which are all potentially responsible for a given observed packet count. It is therefore lower for larger counts since there are fewer flows larger than they. In sketching, ambiguity relates to flows *below* the counter size, since many combinations of smaller flows, or just a single flow, are consistent with a given observed packet count. It is therefore lower at smaller counts as there are far fewer possible combinations.

Technically, the sampling problem is less challenging since sampling methods can be fully characterized by a sampling matrix $\mathbf{B}$, as well as a linear likelihood ($\mathbf{c} = \mathbf{B}\boldsymbol{\theta}$), which together allow the constrained Fisher inverse $\boldsymbol{\mathcal{I}}^+$ to be explicitly calculated, taking a particularly simple form in the case of FS [28]. In the counter sketch the form of the likelihood is highly non-linear, and although $\mathbf{J}$ itself is explicit as well as highly structured (**??**), it is still very difficult to analyze. Instead, to gain insight, bounds were utilized.

To compare the CRLBs of the two methods, we must first renormalize on a per-flow basis by defining:

Flow sampling	$\mathbf{J}_{\rm FS} = \mathbf{J}$	(from (6))
Counter sketch	$\mathbf{J}_{\rm Sk} = A\mathbf{J}/N_f = \mathbf{J}/\alpha$	(from (**??**))

where the renormalization by $1/\alpha$ converts the per-sketch counter information to a per-flow equivalent. The factor inverts to become α when defining the per-flow CRLB $\boldsymbol{\mathcal{I}}^+_{\rm Sk}$ for the sketch, whereas $\boldsymbol{\mathcal{I}}^+_{\rm FS}$ is already per-flow.

To gain a first feeling for the CRLB in a simple case, Figure 1 shows the bound on the standard deviation for each θ_k when there are $W = 5$ different flow sizes and the distribution $\boldsymbol{\theta}$ is close to uniform. The variances are calculated from the analytic inverse of $\boldsymbol{\mathcal{I}}^+_{\rm FS}$ and $\boldsymbol{\mathcal{I}}^+_{\rm Sk}$. In Figure 2 we focus on the variance for θ_1, which is the easiest case for the sketch. We see that broadly speaking the two methods are comparable.

A. Information Comparison

We use the bounds we derived in the previous section to compare $\mathcal{I}^+_{\rm FS}$ and $\mathcal{I}^+_{\rm Sk}$, due to the difficulty in directly evaluating (12). In this subsection α and p_f are free parameters, with resource constraints ignored. However, we assume a shared value of N_f.

Theorem 4: A sufficient condition for $(\mathcal{I}^+_{\rm FS})_{kk} \leq (\mathcal{I}^+_{\rm Sk})_{kk}$ for $k \in [W]$ is $p_f \geq e^{-\alpha}$.

Proof: Follows immediately from (7) and Theorem 3. ■

Theorem 5: A sufficient condition for $(\mathcal{I}^+_{\rm FS})_{kk} \geq (\mathcal{I}^+_{\rm Sk})_{kk}$ for $k \in [W]$ is

$$p_f \leq \min_{k\in[W]} e^{-\alpha} \left[\frac{\theta_k(1-\theta_k)}{h_k(\theta_k, \theta_{k-1}, \ldots, \theta_1, \alpha)} \right],$$

where $h_k(\theta_k, \theta_{k-1}, \ldots, \theta_1, \alpha)$ is polynomial in α with coefficients from products of θ_ℓ, $\forall \ell \leq k$.

Proof: We compare against $(\mathbf{J}^{-1}_{\rm Sk})_{kk} = \frac{1}{\alpha}\sum_{j=1}^{k} c'^2_{k-j} c_j$ (see (12)) by invoking Theorem 2. Both c'_{k-j} and c_j are functions of α, taking respectively the form $e^{\alpha} f_{j-k}(\theta_k, \theta_{k-1}, \ldots, \theta_1, \alpha)$ and $e^{-\alpha} g_k(\theta_k, \theta_{k-1}, \ldots, \theta_1, \alpha)$, and so $(\mathbf{J}^{-1}_{\rm Sk})_{kk} = e^{\alpha} h_k(\theta_k, \theta_{k-1}, \ldots, \theta_1, \alpha)$. The main result follows by comparing against (7). ■

Figure 3 (a) plots the regions from the last two theorems in the (α, p) parameter space for a truncated exponential distribution with $W = 50$. The dashed central 'equality curve' is where $\mathcal{I}^+_{\rm FS} = \mathcal{I}^+_{\rm Sk}$, obtained numerically from the exact expressions. We see that for the kind of sampling rates commonly used, an equivalent value of α is approximately 3.

Figures 3 (b) and (c) explore the bounds' tightness. The curves labeled with θ_k are 'equality curves', where $(\mathcal{I}^+_{\rm FS})_{kk} = (\mathcal{I}^+_{\rm Sk})_{kk}$, obtained numerically from the exact expressions. As k increases from 1 to W, the equality curves fall at different places within the region between the "Suff 1" bound and the "Suff 2" bound (respectively from Theorems 4 and **??**), and in general the highest and lowest curves are close to the respective bounds. This shows that sufficient bounds that apply to all k simultaneously, as the above do, cannot be much tighter, although typically the bounds are loose for any particular k. Figures 3 (c) shows the same is true for the Leipzig–II trace described in the next section.

B. Comparison with Constraints

The goal here is to provide the simplest means by which a parameter equivalence between FS and the counter sketch can be established, in order to learn which method is likely to enjoy the highest information gathering potential in practice. To do so, models of how the methods could be implemented in routers are needed. We do not attempt to specify implementations in detail, nor claim to establish accurate models of cost. Not only is this beyond the scope of the present paper, it would actually be counterproductive, since for both methods, both memory use and computational efficiency can be enhanced in many clever ways. Such enhancements however, constitute the creation of particular "skampling" schemes, effectively leading to an evaluation of one complex hybrid scheme against another. This is contrary to our goals of establishing the fundamental trade-offs of a 'pure' FS approach against a canonical sketch.

We begin by discussing a number of implementation issues to provide some context for the (crude) implementation models which will drive the comparison.

Implementation issues. For each method we will discuss a basic implementation idea, with the aim of maximizing the commonality across the methods. We then argue that these are useful simplifications by discussing how some of the major drawbacks could be addressed. Both methods assume that N_f can be readily measured. This was justified in [28].

Flow sampling. A single hash function $h(x)$ with range $[R]$, acting deterministically on a flow key x, can be used to achieve the FS itself (accept packet if $h(x) \leq p_f R = N$) in a single step (the trajectory sampling idea [4]). The same hash output for accepted packets can also be used to index into the flow table of size N, which must contain a counter to be incremented. Since separate flow records are a defining feature of FS, the table entry must also store the key (once) and check against it (each packet) to detect collisions.

The core issues are the memory cost of keeping the flow key, and the latency cost of checking for collision under heavy load. The flow table can be implemented as set of subtables, each having a simple Bloom filter summary to speed up record search, based on multilevel hash tables for IP packet forwarding [16], [27]. For example, using these technologies, a flow table with 10,000 records, stored in off-chip SRAM with access speeds of 5-10 nsec to enable the processing of back-to-back packets, gives 123 bits per item (including the 10 bits per item for the summaries), thus a total of 154 KB. Schemes which split the counter into a small amount of SRAM for speed (say 9 bits) and a larger amount of DRAM (say 32 bits) [24] to take the high bits of the counter can be used to reduce cost and yet allow line speed updating. Counter Braids [18] may be used to compress the counts.

The average cost of collisions can be kept low by over–dimensioning N, but to account for strings of packets from a single flow arriving back-to-back, we use a cache of on-chip SRAM (with access speeds of 1-2 nsec) for the most recently updated sampled flows [16]. For example, we can have say 10 records, this time with much smaller counters of 6 bits, amounting to 1.1Kbs. The rest of the memory can be used for simple summary Bloom filters which tell where the associated record may be found in off-chip SRAM [16].

Counter sketch. The allocation of packets to sketch entries can be done with a hash function just as above, only with $R = A$. The key difference is that sketch entries have no collision resolution, resulting in considerably reduced complexity. The counters store multiple flows and so need to be larger than for the flow table. Further implementation details can be found in [12]. Optimization of the counter design can be considered as for the flow table.

Implementation models. Observe that the two methods can

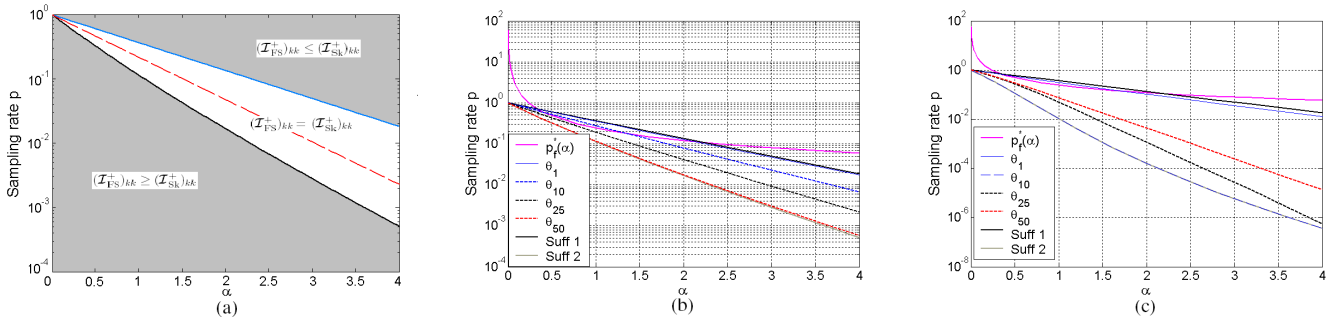

Fig. 3. (a) Illustrating the significance of the results in Theorems 4 and **??**. The dashed line represents the condition $(\boldsymbol{\mathcal{I}}^{+}_{\mathrm{FS}})_{kk} = (\boldsymbol{\mathcal{I}}^{+}_{\mathrm{Sk}})_{kk}$, the blue curve is the upper bound of Theorem 4 (and the upper shaded area) and the black curve is the lower bound of Theorem **??** (and the lower shaded area). The area directly under the dashed line represents $(\boldsymbol{\mathcal{I}}^{+}_{\mathrm{FS}})_{kk} \geq (\boldsymbol{\mathcal{I}}^{+}_{\mathrm{Sk}})_{kk}$, which includes the lower shaded area, and similarly, the area above the dashed line represents $(\boldsymbol{\mathcal{I}}^{+}_{\mathrm{FS}})_{kk} \leq (\boldsymbol{\mathcal{I}}^{+}_{\mathrm{Sk}})_{kk}$, which includes the upper shaded area. The curves were generated using $\boldsymbol{\theta}$ from a truncated exponential distribution with $W = 50$ and $D \approx 16$, and are for the case $k = 30$; (b) and (c), exploring the tightness of the bounds and comparison under constraint (13). Here, $W = 50$ with (b) using the same distribution as (a), and the Leipzig-II trace for (c). The memory constraint curve is $p^*_f = 0.24/\alpha$. Note the log scale of the y–axis.

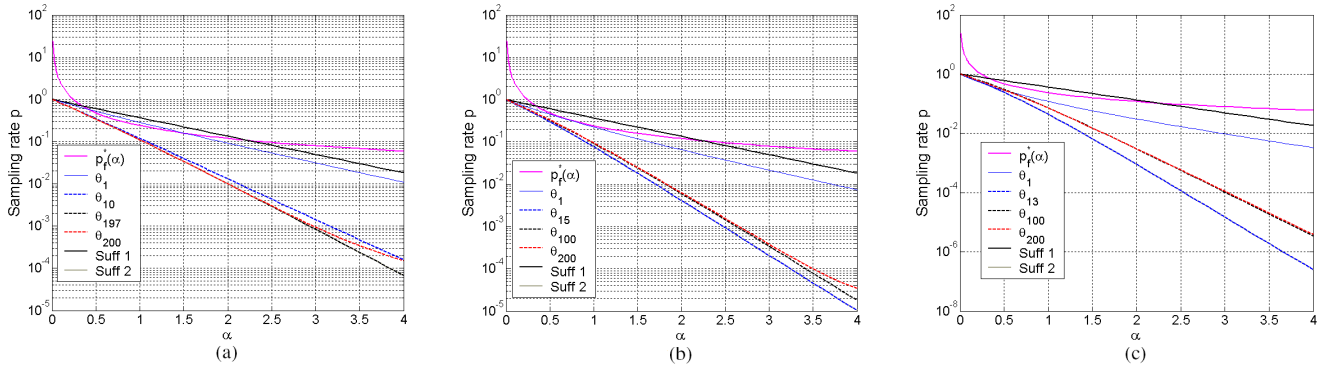

Fig. 4. Comparison under $\boldsymbol{\theta}$ following three Pareto distributions, with $W = 200$, (a) $\rho = 0.5$, $D = 10.869$, (b) $\rho = 1.0$, $D = 1.364$, and (c) $\rho = 2.0$, $D = 3.584$. The memory constraint curve is $p^*_f = 0.24/\alpha$. Note the log scale of the y–axis.

be implemented such that they have significant features in common: a single hash function mapping packets to the data structure, and a single counter for each entry. For simplicity, we will assume that the costs of each of these aspects are identical. This leaves issues associated with collision resolution. The fundamental trade-off is that decreasing p_f reduces these extra costs of FS at the cost of information loss.

Our comparison will be based on the *principle of equal total memory*, being M bits per entry. Let b be the number of bits used in the counter used by either method, and a the number of bits used in the flow key, so that the flow table entry has $a+b$ in total whereas the sketch has only b. In the worst case of uncompressed 5-tuple flow keys, $a = 104$ bits. In practice the number of entries in the flow table must exceed the mean value N by a margin to ensure that collisions are rare. For large N_f this margin does not have to be large in relative terms and we neglect it here. We abstract the traffic data through a flow arrival rate ν_F and the flow size distribution $\boldsymbol{\theta}$, and let α be a free parameter.

All other parameters are now determined: $A = M/b$, $N_f = A\alpha$, $T = N_f/\nu_F$ for the measurement duration, and $N = M/(a+b)$ for the flow table size. Finally, the matching value of p_f is

$$p^*_f(\alpha) = \frac{N}{N_f} = \frac{b}{a+b} \cdot \frac{1}{\alpha}. \tag{15}$$

For example, if $a = 104$ for 5-tuple flows and $b = 32$, the constraint curve is given by $p^*_f = 0.24/\alpha$, shown in Figure 3 (b). The distributions our examples are based on are not an exhaustive study, but close enough to actual traffic to gain some insight on the performance of these methods.

In Figure 3 (b), we compare both methods on $\boldsymbol{\theta}$ following a truncated exponential distribution with $W = 50$ and $D = 16.032$. There is a region where the sketch beats FS for $k \leq 10$. The constraint curve intersects the equality curve for θ_1, showing that there is a region of α values where the sketch outperforms flow sampling for θ_1. This region is much smaller for $k = 10$ compared to $k = 1$ showing that as the flow gets larger, it becomes more difficult for the sketch to outperform FS. Although not shown here, the sketch can outperform FS up to $k = 18$. For $k = 18$, the region is $\alpha \in [0.744, 0.834]$ ($p_f \in [0.288, 0.323]$). In Figure 3 (c), a similar observation applies to an actual trace, Leipzig-II (see Table I for its properties), truncated to $W = 50$, with $D = 1.926$. The sketch's performance is far worse here, since

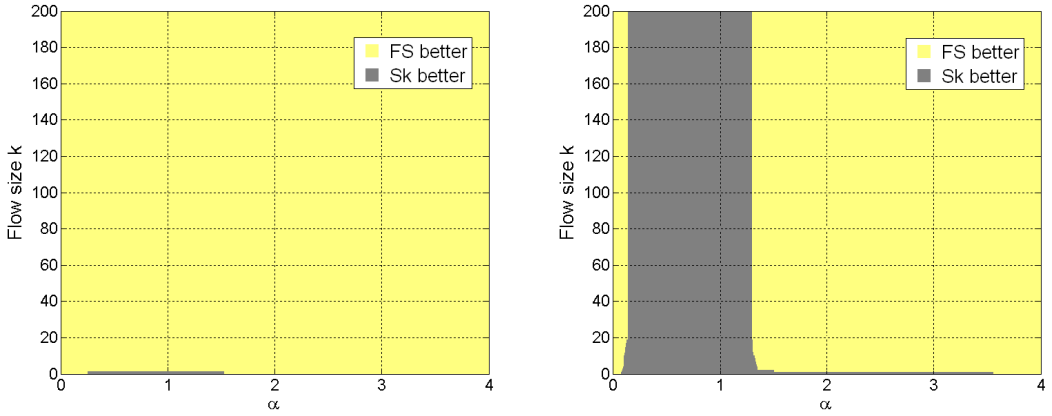

Fig. 5. Performance evaluation under $\boldsymbol{\theta}$ following the Pareto distribution, with $W = 200$, $\rho = 0.5$, $D = 10.869$, to determine the region of flow sizes k where flow sampling dominates the sketch and vice versa. **Left:** Under memory constraints, $p_f^* = 0.24/\alpha$, **Right:** Under cost constraints, $p_f^* = 0.072/\alpha$.

it can only outperform FS in a small α region for $k = 1$. Here, $\alpha \in [0.367, 2.222]$ ($p_f \in [0.108, 0.654]$).

The Pareto distribution is a power law probability distribution determined by two parameters: $t_m > 0$, the minimum value of its support, and $\rho > 0$, the decay rate. Its density is defined as $f(t) = \frac{\rho t_m}{t^{\rho+1}}$ for $t > t_m$ over the support $[t_m, \infty)$. Here, $t_m = 1$ corresponding to the smallest flow size. For our comparisons, we truncated the distribution to $W = 200$. Figures 4 (a) to (c) compares the two methods on three truncated Pareto distributions, with $D = \{10.869, 3.584, 1.364\}$, and $\rho = \{0.5, 1.0, 2.0\}$ respectively. For the last two distributions, although hard to see, the constraint curve lies entirely in the region where FS completely dominates the sketch.

For these distributions, we conclude that FS outperforms the sketch on medium to large flows ($k \geq 20$). As for small flows, FS dominates over a wide range sampling rates p_f, proving that it has much better overall performance. Since these distributions are close approximations to actual traffic, these results hint at FS's potential on actual traffic distributions.

In the simple comparison above, however, the cost of the system was not considered. In practice, SRAM is more expensive than DRAM, and the high usage of SRAM in FS makes it an expensive method. Our comparison will now be based on the *equal cost*, denoted by C_M dollars. We assign weights w_1 and w_2 (units: bit per dollar), $w_1 > w_2$ to account for the cost of SRAM and DRAM respectively. We recompute the costs for each scheme based on their specific SRAM and DRAM usage. The counter array requires 7 bits in SRAM and a 32-bit DRAM backing store per counter [12], hence the cost per counter is $7w_1 + 32w_2$ dollars. We assume FS uses the same amount per counter with an additional 104 bits (using the 5-tuple definition) for collision resolution in SRAM. The cost per flow table entry is therefore $111w_1 + 32w_2$ dollars.

Repeating the calculations, we have $A = C_M/(7w_1 + 32w_2)$, $N_f = A\alpha$, $N = C_M/(111w_1 + 32w_2)$, so we obtain $p_f^*(\alpha) = (7w_1 + 32w_2)/((111w_1 + 32w_2)\alpha)$. Considering that SRAM can be up to 30 times more expensive than DRAM [21], $(w_1, w_2) = (30, 1)$ so $p_f^*(\alpha) = 0.072/\alpha$, with FS at a disadvantage. This time, the range where the sketch outperforms FS is much larger. At large α, however, FS still outperforms the sketch. Clearly, any form of collision resolution should be implemented in DRAM, or at worst, its drawback could be lessened if partial collision resolution was utilized instead.

In Figure 5, we highlight the difference between comparison under pure memory versus cost constraints. We use the same Pareto distribution from Figure 4 (a). We examine the α regions where FS dominates the sketch (the lighter region) and vice versa (the darker region) for flow sizes $k \in [W]$. It can be seen under the pure memory constraint (left figure), FS dominates the sketch, except for $k = 1$ in the region $\alpha \in [0.410, 1.504]$ ($p_f \in [0.16, 0.59]$). Under cost constraints (right), as expected, FS's performance is drastically reduced. It is interesting to point out that in the interval $\alpha \in [0.0862, 1.2879]$ ($p_f \in [0.056, 0.835]$) the sketch will completely dominate for all flow sizes.

To summarize, first, FS is clearly suited for medium and large flows, the caveat being that there must be enough of these flows present for sampling to work. Even for small flows, FS performs well over a large range of sampling rates. Overall, in the pure information theoretic sense, FS is the better option compared to the sketch. Second, the collision resolution mechanism disadvantages FS considerably. From a cost perspective, the counter sketch is competitive with FS. The choice of method for measurement highly depends on the application. For example, anomaly detection would strongly favor the sketch, where a majority of attacking flows are short, while heavy hitter applications would benefit from FS. The sketch is susceptible to high traffic loads, requiring more exports to the collection center to maintain a reasonable α, while FS can adapt by decreasing p_f. Finally, online estimation of the flow size distribution is possible for FS, but not the sketch due to its higher decoding complexity (see Section IV).

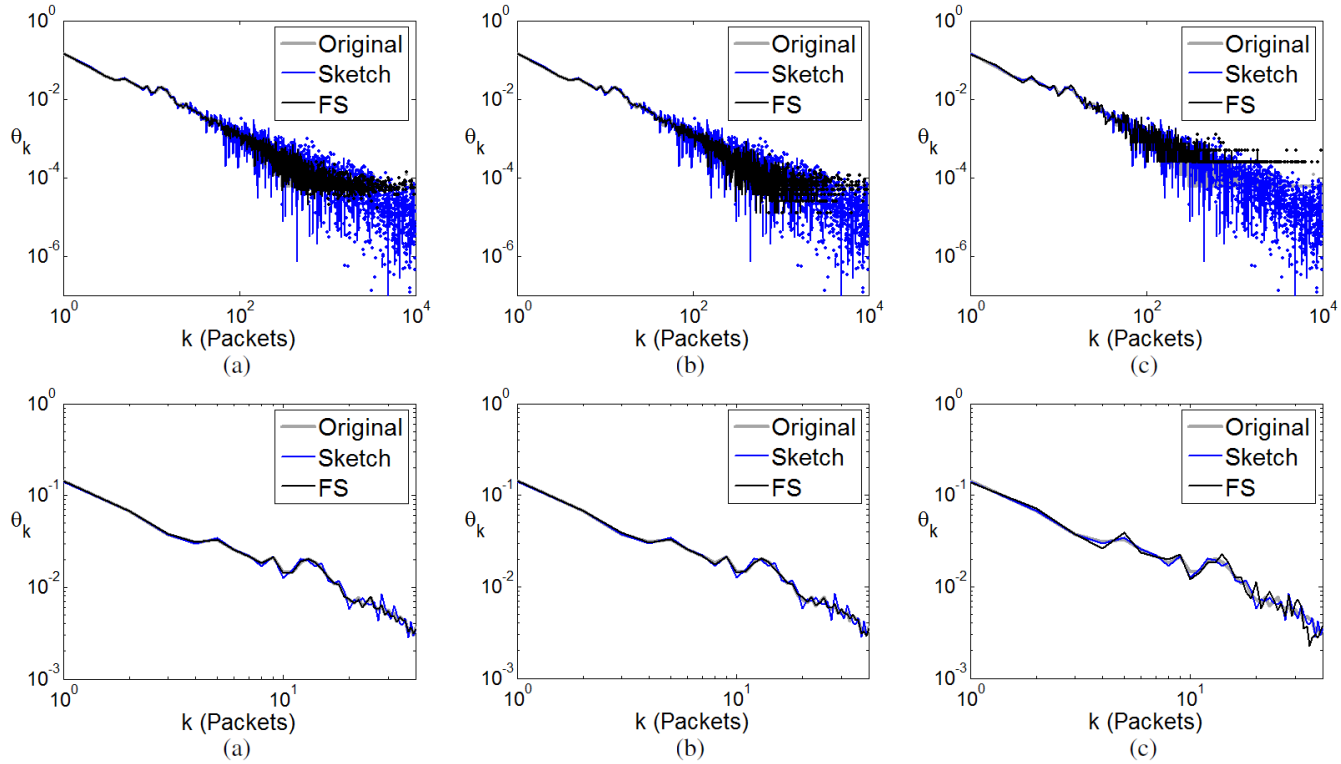

Fig. 6. Comparison between sketch with $\alpha = 4.0$ and FS with (a) $p_f = 0.06$, (b) $p_f = 0.02$ and (c) $p_f = 0.001$, on the distribution Leipzig-II. The sketch estimate is identically the same over each plot, only FS changes. Each plot in the bottom row is a zoom over $1 \leq k \leq 40$ of the plot above.

IV. Data Evaluation

We evaluate both methods using the well–known Leipzig–II [23] trace (see Table I). Since each flow is processed using a hash function without a timeout mechanism, we are in effect assuming an infinite timeout, that is, flows are expired only at the end of the measurement interval (flow splitting is ignored).

Whereas up to now we have focused on Fisher information, our evaluation here will be on estimating flow size distributions themselves, which means we must actually select an estimator for each method. We use maximum likelihood estimators as they are asymptotically unbiased and efficient as $N_f \to \infty$ [11]. We next derive these explicitly for each method.

As it is difficult to visually distinguish the different estimates, we also provide the ℓ_2 distance between the parameter $\boldsymbol{\theta}$ and its estimate $\hat{\boldsymbol{\theta}}$, namely

$$\|\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}\|_2 = \sqrt{\sum_{k=1}^{W} (\hat{\theta}_k - \theta_k)^2}. \tag{16}$$

This is by no means the only way to measure estimation error, however it suffices to quantify how the difference between the methods varies with parameters.

A. Analytic Derivation of the MLEs

Under our framework, we are able to derive the closed–form solutions of the MLE for FS and the counter sketch. The following result is found in [29].

Trace	Link Capacity	Active Flows	Duration
Leipzig-II	50 Mbps	2,277,052	02:46:01

TABLE I
Summary statistics of the data trace.

Theorem 6: The MLE for flow sampling is given by

$$\hat{\boldsymbol{\theta}} = \frac{1}{p_f N_f} [M'_1, M'_2, \cdots, M'_W]^{\mathrm{T}}.$$

where M'_js are the empirical counts of sampled flows of size j. The MLE is unbiased for all sample sizes.

The MLE has a simple interpretation: the estimated flow size distribution is the empirical histogram, appropriately scaled by the sampling parameter p_f.

The MLE for the sketch, however, is more complex, due to the nonlinear nature of the likelihood. Denote by G'_j the number of counters with a packet count of j. The proof is found in Appendix B.

Theorem 7: Let x_k denote the number of flows of size k, $\mathbf{x} = [x_1, x_2, \cdots, x_W]$ be a *flow collision pattern* and Ω_j be the set of flow collision patterns which add up to j. The MLE $\hat{\boldsymbol{\theta}}$ is the solution to the equation,

$$\frac{\bar{\mathbf{c}}}{\mathbf{1}_W^{\mathrm{T}} \bar{\mathbf{c}}} = \frac{1}{\sum_{j=1}^{W} G'_j} [G'_1, G'_2, \cdots, G'_W]^{\mathrm{T}},$$

where $\bar{\mathbf{c}} = \mathbf{diag}(c_1, c_2, \cdots, c_W)(\mathbf{H}^+)^{\mathrm{T}}\mathbf{1}_W$. Note that information on $\boldsymbol{\theta}$ comes only from counters with up to W packet counts. Assuming $W \to \infty$, a closed–form solution is obtained by solving for $\hat{\boldsymbol{\theta}}$ using forward-substitution starting from $\hat{\theta}_1$ (see Appendix B), to yield,

$$\hat{\theta}_1 = \frac{e^{\alpha} - 1}{\alpha(A - G'_0)} \cdot G'_1, \tag{17}$$

and for $2 \leq k < \infty$,

$$\hat{\theta}_k = \frac{e^{\alpha} - 1}{\alpha(A - G'_0)} \cdot G'_k - \sum_{\mathbf{x} \in \Omega_k} \prod_{\ell=1}^{k-1} \frac{\alpha^{x_\ell - 1}}{x_\ell!} \hat{\theta}_\ell^{x_\ell}.$$

In the examples below we use this infinite W closed form result, details of the evaluation of which are given in Appendix C. This is justified by the large value of W we used in the trace analysis.

B. Method Comparison

We compare the two methods using $\alpha = 4.0$ with $A = 569,263$ counters for the Leipzig trace. The average flow arrival rate is $\nu_F = 229$ flows/sec. Large α values such as this can arise under high transient loads, or through longer measurement intervals designed to better utilize available sketch memory. The left plot of Figure 5 can be used as a rough guide to the relative performance we can expect here, since the Pareto distribution is not a bad match to the true flow size distribution. It indicates that with $\alpha = 4$ FS should dominate at all flow sizes under a fair memory comparison.

We plot three comparisons in Figures 6 (a) to (c) with $W = 10,000$ and rates $p_f = \{0.06, 0.02, 0.001\}$, and give corresponding zooms on the front of the distribution in the plots (d) to (f). In all these the same sketch estimate is plotted, only the FS curve changes. We stress that these comparisons are not directly related to the results in Section III. Here, we compare a single estimate of the flow size distribution for each scheme, rather than a bound on the estimation variance. Nonetheless, we expect the relative performance to roughly obey the results from previous sections. For a more direct comparison one can compute the *observed Fisher information* [20] and compare it with the exact Fisher information. We elaborate on this further below.

We chose $p_f = 0.06$ for Figure 6 (a) as this corresponds to the standard 5-tuple definition under the equal memory constraint: $p_f^* = 0.24/\alpha = 0.06$. With this choice, FS is expected to outperform the sketch (see Figure 4 (b) as a guide). The value $p_f = 0.02$ is slightly larger than 0.0183, when Theorem 4 for the superiority of FS is just met. We therefore again expect FS to dominate. This is very close to the rate $p_f^* = 0.072/\alpha = 0.018$ corresponding to a memory cost comparison. Finally, $p_f = 0.001$ is when Theorem **??** applies and the sketch is predicted to do better. The comparison however is no longer memory or cost fair in this case.

The main observation from Figure 6 is that each scheme is performing well, especially at the front end of the distribution as clearly seen in the zooms. As expected, FS performs less well as p_f drops from plot (a) through (b) to (c). Moreover, the results agree with the predictions from theory just quoted, with FS being superior to the sketch in (a) and (b), but worse in (c). This is visible (albeit with difficulty) in the plots, and is confirmed by ℓ_2 errors for FS with $p_f = \{0.06, 0.02, 0.001\}$ of $\{0.0020, 0.0030, 0.015\}$ respectively, compared to 0.013 for the sketch. However, the crossover value of p_f where the schemes are equal is smaller than predicted by theory, that is FS is performing a bit better in practice than expected.

The schemes differ in their tail behavior. For FS, the estimates overlay the true distribution well until they become erratic due to lack of samples (which of course occurs earlier for smaller p_f), and errors are one-sided, on the too-small side, since some long flows are missed but false long flows are not created. In contrast the sketch fails to follow the detail of the tail after around $k = 100$, and exhibits a two sided error (which increases in magnitude with k) because of the different nature of its inversion. Notwithstanding the above, note that in the 'tail' here ($k \geq 100$) it is not possible to draw clear conclusions on relative performance because of the high variability inherent in single 'point' estimates.

We now turn our attention to the observed Fisher information, whose inverse is the observed CRLB. For the distribution above, the observed CRLB of θ_1 for the sketch was 7.67, while it is $\{2.02, 6.12, 120.04\}$ for $p_f = \{0.06, 0.02, 0.001\}$ respectively for FS, in rough agreement with the predictions for these p_f values as described above. Such estimates are difficult to perform with large W however, and doing so is part of ongoing work. While not shown here, the methods were tested on the Auckland–II trace [8], with closer agreement to theory due to the larger size of the trace. Finally, the results here do not compare to the empirical results of [12], since they compare against packet sampling, not FS.

V. Conclusions

We have presented for the first time an information theoretic comparison between sampling and sketching, in the context of the measurement of the flow size distribution. The rigorous results enable the different natures of the inherent ambiguity of each approach to be better understood, and the potential for information gathering to be compared rigorously. We found that the two approaches were largely comparable in terms of information, and because of the possibility of a hash based sampling operation, were also much closer computationally than has previously been supposed. FS has many strengths, namely its flexibility with regards to traffic load variations and better estimation of medium to large flows (provided adequate samples are obtainable). From an economic perspective, however, FS is much more expensive, due to its requirement of high speed collision resolution mechanisms. In this regard, the counter sketch is more economically viable, although, information theoretically, its performance for medium and large flows is less optimal compared to FS.

APPENDIX

A. Two Important Properties

The two important properties for us are:

1) $\mathbf{H}^{+}\mathbf{H} = \mathbf{I}_W$ and $(\mathbf{H}^{\mathrm{T}})(\mathbf{H}^{\mathrm{T}})^{+} = \mathbf{I}_W$, and
2) the products $\mathbf{H}\mathbf{H}^{+}$ and $(\mathbf{H}^{\mathrm{T}})^{+}(\mathbf{H}^{\mathrm{T}})$ result in the matrix

$$\begin{bmatrix} \mathbf{I}_W & \mathbf{0}_{W\times W} & \cdots \\ \mathbf{0}_{W\times W} & \mathbf{0}_{W\times W} & \cdots \\ \vdots & \vdots & \ddots \end{bmatrix}$$

The second property is used in deriving the MLE (see below).

B. Proof of Theorem 6

For the counter sketch, the likelihood function for N_f flows is

$$f(\boldsymbol{\theta}, N_f) = \prod_i f(j_i;\boldsymbol{\theta}) = \prod_{j=0}^{\infty} c_j^{G'_j},$$

where G'_j represents the empirical number of counters with a packet count of j. The MLE is the $\boldsymbol{\theta}$ which maximizes the log-likelihood

$$l(\boldsymbol{\theta}, N_f) = \sum_{j=0}^{\infty} G'_j \log c_j$$

subject to the constraint $\sum_{k\geq 1}\theta_k = 1,\ \theta_k > 0,\ \forall k \in [W]$. A feasible solution always exists by the Bolzano–Weierstrass theorem [**?**, p. 517], since the log-likelihood function is concave and continuous, and optimization is performed over a compact, convex set. Our assumptions ensure the solution obtained will be unique, since the Hessian of the log-likelihood is the Fisher information, which is positive definite given $0 < \theta_k < 1$ for all k.

By our assumptions on the constraints, the method of Lagrange multipliers yields the optimal solution by strong duality since the problem satisfies Slater's constraint qualification [**?**, Section 5.2.3, p. 226]. The Lagrangian is

$$\mathcal{L}(\boldsymbol{\theta}, \mu, \boldsymbol{\nu}) = \sum_{j=0}^{\infty} G'_j \log c_j - \mu\Big(\sum_{k\geq 1}\theta_k - 1\Big) - \boldsymbol{\nu}^{\mathrm{T}}\boldsymbol{\theta},$$

where the vector $\boldsymbol{\nu}$ has elements $\nu_k \geq 0$ and $\mu \in \mathbb{R}$. By differentiating with respect to $\theta_k,\ \forall k$, and assuming $\boldsymbol{\theta}$ lies in the interior of $[0,1]^W$ so that $\boldsymbol{\nu} = \mathbf{0}_{W\times 1}$, we have

$$\alpha\mathbf{H}^{\mathrm{T}}\mathbf{diag}(c_1^{-1}, c_2^{-1}, \cdots)\mathbf{diag}(G'_1, G'_2, \cdots)\mathbf{1}_\infty = \mu\mathbf{1}_W, \quad (18)$$

Recall that

$$\mathbf{h} = \left[\sum_{k=0}^{W-1} c'_k, \sum_{k=0}^{W-2} c'_k, \cdots, c'_0, 0, 0, \cdots\right]^{\mathrm{T}},$$

with the $W+1$-th term and onwards are zero. We solve for μ by taking the pseudo-inverse of $\mathbf{H}^{\mathrm{T}}$ on both sides, and noting property (ii) in Appendix A, results in terms $W+1$ onwards on both sides of the equation to equal 0. We then only need to restrict our attention to the first W terms, obtaining

$$\mu\mathbf{h}_W = \alpha\mathbf{diag}(c_1^{-1}, c_2^{-1}, \cdots, c_W^{-1})\mathbf{diag}(G'_1, G'_2, \cdots, G'_W)\mathbf{1}_W.$$

By multiplying $\mathbf{1}_W^{\mathrm{T}}\mathbf{diag}(c_1, c_2, \cdots, c_W)$ on both sides of the equation, and rearranging, we have

$$\mu = \frac{\alpha(\sum_{j=1}^{W} G'_j)}{\mathbf{1}_W^{\mathrm{T}}\bar{\mathbf{c}}}. \quad (19)$$

Rewriting (16),

$$\mathbf{H}^{\mathrm{T}}\mathbf{diag}(c_1^{-1}, c_2^{-1}, \cdots)\mathbf{diag}(G'_1, G'_2, \cdots)\mathbf{1}_\infty = \frac{\alpha(\sum_{j=1}^{W} G'_j)}{\mathbf{1}_W^{\mathrm{T}}\bar{\mathbf{c}}}\mathbf{1}_W,$$

and multiplying by $\mathbf{diag}(c_1, c_2, \cdots, c_W, 0, 0, \cdots)(\mathbf{H}^{\mathrm{T}})^{+}$ on both sides, and, since terms from $W+1$ onwards are zero, we solve the equation by focussing on the first W to obtain

$$(\sum_{j=1}^{W} G'_j)\cdot\frac{\bar{\mathbf{c}}}{\mathbf{1}_W^{\mathrm{T}}\bar{\mathbf{c}}} = [G'_1, G'_1, \cdots, G'_W]^{\mathrm{T}},$$

leading to the result.

If we assume instead that $W \to \infty$, (16) becomes

$$\mathbf{H}^{\mathrm{T}}\mathbf{diag}(c_1^{-1}, c_2^{-1}, \ldots)\mathbf{diag}(G'_1, G'_2, \ldots)\mathbf{1}_\infty = \frac{A - G'_0}{1 - e^{-\alpha}}\mathbf{1}_\infty.$$

Each estimate $\hat{\theta}_k$ for $k = 1, 2, \ldots$ is solved by forward-substitution beginning from case $k = 1$, leading to

$$\hat{\theta}_k = \frac{e^{\alpha}}{\alpha}\cdot\frac{1 - e^{-\alpha}}{A - G'_0}G'_k - \sum_{\mathbf{x}\in\Omega_k}\prod_{\ell=1}^{k-1}\frac{\alpha^{x_\ell - 1}}{x_\ell!}\hat{\theta}_\ell^{x_\ell}.$$

C. Numerical Evaluation of the Sketch MLE

In the evaluation of the closed form formula above for the sketch MLE, the difficulty lies in the computation of the second, negative, term, which is crucial for collision resolution.

We start with the estimate $\hat{\theta}_1$, given by equation (15). For the rest, in order to compute the negative term for $\hat{\theta}_k$, we need to compute all possible combinations of flow collisions that add up to a packet counter value of k. To do this, we first generate the sequences $S_j = \{\alpha^m\hat{\theta}_j^m/m!\}_{m=0}^{d}$, for some integer $0 < d < \infty$ and $j = 1, 2, \ldots, k-1$ (in practice, we store and reuse results across different k). Then, we compute the 'total convolved' sequence $T_k = S_1 * U_2(S_2) * \cdots * U_{k-1}(S_{k-1})$, where $U_i(\cdot)$ denotes the operator that upsamples a sequence by a factor i.

The $k+1$-th element in T_k, after division by α, is just the negative term. Let

$$S_j^*(z) = \sum_{m_j=0}^{d}\frac{\alpha^{m_j}\hat{\theta}_j^{m_j}}{m_j!}z^{m_j}$$

be the generating function for the sequence S_j and we assume $|z| < 1$ for convergence of the function. Then the 'total

convolved' generating function is

$$\begin{aligned} T_k^*(z) &= S_1^*(z) \cdot S_2^*(z^2) \cdot \ldots \cdot S_{k-1}^*(z^{k-1}) \\ &= \sum_{m_1=0}^{d} \frac{\alpha^{m_1}\hat{\theta}_1^{m_1}}{m_1!} z^{m_1} \cdot \sum_{m_2=0}^{d} \frac{\alpha^{m_2}\hat{\theta}_2^{m_2}}{m_2!} z^{2m_2} \cdot \ldots \\ &\quad \cdot \sum_{m_{k-1}=0}^{d} \frac{\alpha^{m_{k-1}}\hat{\theta}_{k-1}^{m_{k-1}}}{m_{k-1}!} z^{(k-1)m_{k-1}} \\ &= \sum_{m_1=0}^{d} \cdots \sum_{m_{k-1}=0}^{d} \prod_{j=1}^{k-1} \frac{\alpha^{m_j}\hat{\theta}_j^{m_j}}{m_j!} z^{m_1+\ldots+(k-1)m_{k-1}} \\ &= \sum_{n=0}^{d^{k-1}} \left(\sum_{\mathbf{x}\in\bar{\Omega}_n} \prod_{\ell=1}^{n-1} \frac{\alpha^{x_\ell}}{x_\ell!} \hat{\theta}_\ell^{x_\ell} \right) z^n. \end{aligned}$$

In this way, $m_1, m_2, \ldots, m_{k-1}$ covers collision patterns for packet counter values from 0 to d^{k-1}, which can be rewritten as the expression in the last line. In the last line, $\bar{\Omega}_k \subseteq \Omega_k$ is a restricted set of possible collision patterns, since $d < \infty$. Notice that combinations of $m_1, m_2, \ldots, m_{k-1}$ that sum to k, divided by α, would be equivalent to the negative term for $\hat{\theta}_k$, which is just the coefficient of the term $n = k$ in the last line. Thus, the negative term for $\hat{\theta}_k$ is simply equal to the $k+1$-th term of the sequence T_k.

Note that d here represents the maximum sequence length, which in the Poisson approximation we use, should be ∞ to account for all possible collisions. Clearly, however, this is not computationally feasible, and in our implementation, we set $d = 10$, noting that the decay of $\alpha^m \hat{\theta}_j^m / m!$ is exponential in m, since $\hat{\theta}_j < 1$, $\forall j \in [W]$. We also find this to be a good tradeoff against computational error, the result of accumulative error from convolving many sequences.

A large chunk of the computational complexity lies in the convolution of the sequences S_j. To optimize the estimation procedure, a good convolution algorithm is required. In our implementation, we performed convolution in the frequency domain, by using a Fast Fourier transform, followed by an inverse Fourier transform to obtain the result. This was done by the `fftfilt` function in Matlab. In terms of space complexity, the main drawback is the storage of T_k, which can be very large. A subject of future work is to optimize memory usage.

REFERENCES

[1] Cisco NetFlow. http://www.cisco.com/en/US/products/ps6601/ products_ios_protocol_group_home.html.

[2] N. Alon, Y. Matias, and M. Szegedy. The space complexity of approximating the frequency moments. In *STOC '96*, pages 20–29, New York, NY, USA, 1996. ACM.

[3] B. H. Bloom. Space-time trade-offs in hash coding with allowable errors. *Communications of the ACM*, 13(7):422–426, 1970.

[4] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

[5] N. Duffield and M. Grossglauser. Trajectory sampling for direct observation. *IEEE/ACM Trans. Networking*, 9(3):280–292, June 2001.

[6] N. Duffield, C. Lund, and M. Thorup. Estimating flow distributions from sampled flow statistics. *IEEE/ACM Trans. Net.*, 13(5):933–946, 2005.

[7] C. Estan and G. Varghese. New directions in traffic measurement and accounting. *ACM Transactions on Computer Systems*, 21(3):270–313, August 2003.

[8] J. D. Gorman and A. O. Hero. Lower bounds for parametric estimation with constraints. *IEEE Trans. Info. Theory*, 36(6):1285–1301, November 1990.

[9] W. R. Group. Auckland-II Trace Data. http://pma.nlanr.net/Traces/long/auck2.html.

[10] D. Harville. *Matrix Algebra from a Statistician's Perspective*. Springer-Verlag, 1997.

[11] N. Hohn and D. Veitch. Inverting sampled traffic. *IEEE/ACM Trans. Net.*, 14(1):68–80, February 2006.

[12] S. M. Kay. *Fundamentals of Statistical Signal Processing, Volume I: Estimation Theory*. Prentice Hall PTR, March 1993.

[13] A. Kumar, M. Sung, J. Xu, and J. Wang. Data streaming algorithms for efficient and accurate estimation of flow size distribution. In *Proc. of ACM SIGMETRICS 2004*, New York, June 2004.

[14] A. Kumar, M. Sung, J. Xu, and E. W. Zegura. A data streaming algorithm for estimating subpopulation flow size distribution. *SIGMETRICS Perform. Eval. Rev.*, 33(1):61–72, 2005.

[15] A. Kumar and J. Xu. Sketch Guided Sampling: Using online estimates of flow size for adaptive data collection. In *Proc. INFOCOM '06*, Barcelona, Spain, April 2006.

[16] A. Kumar, J. Xu, and J. Wang. Space-Code Bloom filter for efficient per-flow traffic measurement. *IEEE JSAC Special Issue on Sampling the Internet: Techniques and Application*, 24(12):2327–2339, December 2006.

[17] S. Kumar, J. Turner, and P. Crowley. Peacock hashing: Deterministic and updatable hashing for high performance networking. In *INFOCOM 2008*, Phoenix, AZ, USA, April 2008.

[18] A. Lall, M. Ogihara, and J. Xu. An efficient algorithm for measuring medium–to large–sized flows in network traffic. In *Proc. INFOCOM 2009*, pages 1–5, April 2009.

[19] Y. Lu, S. Dharmapurikar, A. K. Kabbani, A. Montanari, and B. Prabhakar. Counter Braids: A novel counter architecture for per-flow measurement. In *Proceedings of ACM SIGMETRICS '08*, pages 121–132, June 2008.

[20] J. Mai, C.-N. Chuah, A. Sridharan, T. Ye, and H. Zang. Is sampled data sufficient for anomaly detection? In *ACM IMC '06*, Rio de Janeiro, Brazil, October 2006.

[21] S. Mueller. *Upgrading and repairing PCs*. Que Publishing, 18th edition, 2008.

[22] M. Muthukrishnan. Data streams: Algorithms and applications. *Foundations and Trends in Theoretical Computer Science, 1, 2*, 2005.

[23] NLANR. Leipzig-II Trace Data. http://pma.nlanr.net/Special/leip2.html.

[24] S. Ramabhadran and G. Varghese. Efficient implementation of a statistics counter architecture. *ACM SIGMETRICS Performance Evaluation Review*, 31(1):261–271, June 2003.

[25] B. Ribeiro, D. Towsley, T. Ye, and J. Bolot. Fisher information of sampled packets: an application to flow size estimation. In *Proceedings of the 6th ACM SIGCOMM on Internet Measurement*, pages 15–26, Rio de Janeiro, Brazil, October 2006.

[26] B. Ribeiro, T. Ye, and D. Towsley. A resource minimalist flow size histogram estimator. In *Proc. 2008 ACM SIGCOMM Internet Measurement Conference*, pages 285–290, Vouliagmeni, Greece, October 2008.

[27] H. Song, S. Dharmapurikar, J. Turner, and J. Lockwood. Fash hash table lookup using extended Bloom filter: an aid to network processing. In *SIGCOMM 2005*, Philadelphia, PA, USA, August 2005.

[28] A. E. Taylor and W. R. Mann. *Advanced Calculus*. John Wiley and Sons, 3rd edition, 1983.

[29] P. Tune and D. Veitch. Towards optimal sampling for flow size estimation. In *Proceedings of the 8th ACM SIGCOMM on Internet Measurement*, pages 243–255, Vouliagmeni, Greece, October 2008.

[30] P. Tune and D. Veitch. Fisher information in flow size distribution estimation. Technical Report TR08-001, University of Melbourne, 2009. Available at http://www.cubinlab.ee.unimelb.edu.au/∼lsptune/.

[31] R. Zamir. A proof of the Fisher information inequality via a data processing argument. *IEEE Trans. Info. Theory*, 44(3):1246–1250, May 1998.

[32] H. Zhao, A. Lall, M. Ogihara, O. Spatscheck, J. Wang, and J. Xu. A data streaming algorithm for estimating entropies of OD flows. In *IMC '07*, pages 279–290. ACM, 2007.