

Estimating Flow Distributions from Sampled Flow Statistics

Nick Duffield
AT&T Labs—Research
180 Park Avenue
Florham Park, NJ 07932, USA
duffield@research.att.com

Carsten Lund
AT&T Labs—Research
180 Park Avenue
Florham Park, NJ 07932, USA
lund@research.att.com

Mikkel Thorup
AT&T Labs—Research
180 Park Avenue
Florham Park, NJ 07932, USA
mthorup@research.att.com

ABSTRACT

Passive traffic measurement increasingly employs sampling at the packet level. Many high-end routers form flow statistics from a sampled substream of packets. Sampling is necessary in order to control the consumption of resources by the measurement operations. However, knowledge of the statistics of flows in the *unsampled* stream remains useful, for understanding both characteristics of source traffic, and consumption of resources in the network.

This paper provide methods that use flow statistics formed from sampled packet stream to infer the absolute frequencies of lengths of flows in the unsampled stream. A key part of our work is inferring the numbers and lengths of flows of original traffic that evaded sampling altogether. We achieve this through statistical inference, and by exploiting protocol level detail reported in flow records. The method has applications to detection and characterization of network attacks: we show how to estimate, from sampled flow statistics, the number of compromised hosts that are sending attack traffic past the measurement point. We also investigate the impact on our results of different implementations of packet sampling.

Categories and Subject Descriptors

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

General Terms

Measurement, Theory

Keywords

Packet Sampling, IP Flows, Maximum Likelihood Estimation

1. INTRODUCTION

1.1 Motivation and Challenges

Passive traffic measurement increasingly employs sampling at the packet level to control the consumption of resources in measurement subsystems and infrastructure. As a first example, many

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

SIGCOMM'03, August 25–29, 2003, Karlsruhe, Germany.
Copyright 2003 ACM 1-58113-735-4/03/0008 ...\$5.00.

high end routers form flow statistics from only a sampled substream of packets in order to limit the consumption of memory and processing cycles involved in flow cache lookups. As a side benefit, the rate at which flow statistics are produced is reduced, in most cases, lowering the requirement for bandwidth to transmit flow statistics to a collector, and for processing and storage costs at the collector. As a second example, reports on individual packets are exported from a router to a collector. Keeping a record of every packet in the network is infeasible: packet sampling at the router is necessary to control usage of processing resources, bandwidth to the collector, and processing and storage costs at the collector.

Sampling entails an inherent loss of information. For some purposes, loss is easy to correct for. Assuming that 1 in N packets are selected on average, the total number of packets in the stream can be estimated by multiplying the number of sampled packets by N . Assuming sampling decisions to be independent of packet size, the total number of bytes can be estimated in the same way.

However, more detailed characteristics of the original traffic are not so easily estimated. Quantities of interest include the number of packets in the flow—we shall refer to this as the flow length—and the total bytes that those packets contain. When packet sampling is employed in routers, the measurements reported are those for the sampled packet stream rather than the original packet stream. We call the statistics so formed sampled flow statistics. What relation do the sampled flow statistics bear to the flow statistics of the original unsampled packet stream? Some original flows will not be sampled at all, and longer flows are more likely to be sampled than shorter ones. Thus simply scaling all sampled flow lengths by N will not give a good estimate of the number of original flows, or the distribution of their lengths.

Knowing the number and lengths of the unsampled flows remains useful characterizing traffic and the resources required to accommodate its demands. Here are some applications:

Resources Required for Collecting Flow Statistics: flow cache utilization and the bandwidth for processing and transmitting flow statistics are sensitive to the sampling rate, the number of flows, and flow lengths and duration; see [8, 9].

Characterizing Source Traffic: the measured numbers of flows and the distribution of their lengths have been used to evaluate gains in deployment of web proxies [11], and to determine thresholds for setting up connections in flow-switched networks [12].

Characterizing Network Attacks: in particular, estimating the number of hosts generating the attack traffic in a set of sampled flow statistics. This will be an application of our method in Section 8.

Although sampled traffic statistics are increasingly being used for network measurements, to our knowledge no studies have ad-

dressed the problem of estimating the characteristics of flows in the original unsampled packet stream—in particular the frequencies at which different numbers of packets per flow occur—from the same characteristics of flows constructed from the sampled packet stream. This is the topic addressed in this paper. For the applications described above, we envisage packet sampled flow statistics would be either be constructed directly at routers, or formed at a collector by aggregation of reports on individual sampled packets collected by a router that forms no flow statistics itself.

Since sampling picks on average 1 in N packets from an original flow, it is tempting to propose the following simple scaling argument: attribute to each sampled flow of length ℓ an original flow of length $N\ell$. While simple to implement, this approach has a number of drawbacks. First, it takes no account of flows that have none of their packets sampled, and so the total number of original flows is undercounted. Specifically, the inferred frequencies of the original flows are biased against (i.e. undercount) shorter flows, since these are less likely to be sampled. Although it is possible to compensate any inference against this bias (and we do so later in the paper) a second drawback remains: the inferred distribution of flow lengths would be concentrated on integer multiples of N . In practice, measured flow length distributions are smoother, so some effective manner of smoothing would be required. This need is particularly evident for small flow lengths. When N is large, much of the detail of the original flow length distribution may be at lengths much shorter than N , the shortest length that would be inferred from simple scaling. We need to resolve this detail.

With either independent or periodic sampling, flow with lengths far shorter than N will usually have at most one packet sampled. Hence it is a challenge to use the information contained in the frequencies of sampled flow lengths to resolve what details one can of the distribution of original flows shorter than N . Consider the problem of trying distinguish the following sets of original flows from their sampled counterparts: (i) 2,000,000 flows of size 1, and (ii) 1,000,000 original flows of size 2, separately subject to 1 in N packet sampling. The expected number of sampled packets is the same in each case. When N is as large as 10,000, the mean number of sampled flows of size 1 is 200 (to the nearest integer) in each case, and there is only a 1% chance that case (ii) yields *any* sampled flows of length 2. This demonstrates that large differences in the frequencies of original flows can be difficult to distinguish on the basis of the frequencies of the sampled flows alone: further information on the flows is needed to distinguish such cases.

With sampling period $N = 100$, there is only a 1% difference between cases (i) and (ii) in the number of sampled flows of size 1, but there are now an appreciable number of sampled flows of length 2 in case (ii), 100 on average. Any inference method that is to distinguish the frequencies of short original flows must therefore estimate each original frequency as a function of a *set* of sampled frequencies, rather than simply scaling a single frequency.

We expect knowledge of inferred frequencies to be limited to smoothed versions. Consider two original flow length distributions identical except that one is supported on even lengths, the other on odd, i.e., one of the distributions can be obtained by shifting the other one place to the right. From the above arguments it is evident that for sufficiently large sampling period, the resulting sampled flow length distributions will be indistinguishable. Thus, the best we can hope to do is inferred some smoothed set of frequencies.

1.2 Contribution and Outline

The work of this paper meets these challenges using three approaches. The first formalizes the above scaling argument and shows in particular how to smooth the distribution so as to more

accurately predict the distributions of flow lengths shorter than N . The second uses maximum likelihood estimation and associated techniques to estimate the full distribution of packet and byte lengths. The third uses protocol level detail commonly reported in flow statistics (specifically, TCP flags, when available) to supplement the flow level information and render the estimators more accurate.

Section 2 describes the sampling model, the use of protocol level information to supplement sampled flow length statistics, and complexities that arise when multiple measured flows arise from an original flow. In Section 3 we address another question: to what extent do the details of the sampling process affect the sampled flow length distribution, and our ability to infer? Two different implementations of sampling with the same rate—e.g. periodic and independent random sampling—will select different individual packets. However, we find in practice that the distributions of flow lengths that they produce are quite similar. If the differences are ignorable, two useful conclusions may be drawn. First, the implementation details of these two types of packet sampling are relatively unimportant as far as flow length distributions are concerned. This helps foster uniform interpretation of sampled flow lengths across different vendor implementations. Second, when modeling the sampling process in this paper, we are at liberty to chose the implementation which is most convenient for computational purposes.

In Section 4 we show how protocol level detail in the flows can be used to resolve detail of the frequencies of small flow lengths with the scaling based estimator. Section 5 briefly describes a moment based estimator, which while having bad statistical properties itself, is useful in understanding our second main method: a Maximum Likelihood estimator implemented with the Expectation Maximization (EM) algorithm, presented in Section 6. Both estimators are evaluated against packet and flow traces in Section 7.

Lastly, the class of inference problems solved here have another networking application. By letting the variable standing for flow length instead represent the number of packets produced by a compromised host during a certain network attack we can use sampled flow statistics to infer the total number of compromised hosts that sent traffic to a network, i.e., including those from which no attack packets were sampled. This is described in Section 8. We conclude with some proposals for further work in Section 9.

1.3 Related Work

The work most closely related to this paper is [9], which raised the idea of inferring properties of original flows, specifically the mean flow length, from packet sampled flow statistics. The current paper goes much further: we infer the complete distribution of flow lengths. In this paper, the packet sampling model reflects current practice: packets are sampled with some average probability p . Other recent work has proposed a different packet sampling scheme in order to better capture the statistics of longer flows [10]. Adjustment of the sampling rate in order to meet constraints on estimation accuracy was proposed in [2]. The work of [5] concerned a different problem: the efficient estimation of the distribution of packet sizes under sampling. Independent and periodic 1 in N sampling, as well as stratified sampling out of finitely many bins, were compared. The problem of estimating the number of distinct classes in a population from the distribution of class frequencies in a sample has been considered in [13]. In the current setting, this corresponds to estimating the total number of original flows. However, these estimators perform poorly in our application. We investigate the differences between flow length frequencies arising from random and periodic sampling. Discrepancy measures for fitting measured distribution to models have been considered in [5] and [19].

2. FLOWS, SAMPLING & INFORMATION

2.1 The Formation of Flow Statistics

An IP flow is a set of packets, observed in the network within some time period, that share a common key. An example is the “raw” flows observed at a router, where the flow key distinguishes individual source and destination IP address, and TCP/UDP port numbers. In order to compile flow statistics, the router maintains a table of records indexed by flow key. A flow is said to be active at a given time if there exists a record for its key. When a packet arrives at the router, the router determines if a flow is active for the packet’s key. If not, it instantiates a new record for the packet’s key. The statistics for the flow are updated for the packet, typically including counters for packets and bytes, arrival times of the first and most recent packet of the flow.

Flow statistics can be thought of as summarizing application level transactions. However, the router does not assume knowledge of application level flow structure, in particular when the flow has ended. Instead, the router must terminate flows, by criteria that may include: (i) interpacket timeout: the time since the last packet observed for the flow exceeds some threshold; (ii) protocol: e.g., observation a FIN or RST packet of the Transmission Control Protocol (TCP) [22]; (iii) memory management: releasing memory for new flows; (iv) aging: to prevent data staleness, flows are terminated after a given elapsed time since the arrival of the first packet of the flow. When the flow is terminated, its statistics are exported, and the associated memory is released for use by new flows.

Flow definition schemes have been developed in research environments, see e.g. [1, 4], and are being standardized [16]. Reported flow statistics typically include the elements of the key, the arrival times of the first and last packets, and the number of packets and bytes in the flow. Flow statistics are commonly produced using Cisco’s NetFlow [3]. In Inmon’s sFlow [14], reports on sampled packets are exported from routers to a collector. Packet sampling capabilities for routers are currently being standardized [21]. In this context, aggregation of sampled packet reports into flow statistics could be performed in the collection system.

2.2 Flow Semantics and Sampling

A good definition of a flow should encapsulate each application transaction through the flow summary. However, two factors hinder the effectiveness of such encapsulation. First, new applications may generate packets in patterns that are not well captured by the flow definitions. Second, packet sampling removes cues for flow delineation from the packet stream. The FIN packet marking the end of a TCP connection may not be sampled. Interpacket timeout is expected to become the dominant method of termination for TCP flows when the sampling rate is low. We will use the term *original flow* to describe a set of application level packet grouped independently of any specific termination rule used by routers. Once a measurement mechanism has been defined, we can speak of a *measured flow*. Either type of flow can be called *sampled*; for an original flow this means a substream of packets sampled from it, while a sampled measured flow means a flow measured from such a substream.

2.3 Dependence on the Sampling Model

Within the functional requirement of sampling packets at a given rate, a number of different implementations are possible. Implementations include independent sampling of packets with probability $1/N$, and periodic selection of every N^{th} packet from the full packet stream. In both cases we will call N the *sampling period*, i.e., the reciprocal of the average sampling rate. To what extent would the distributions of sampled flow lengths be expected to dif-

fer, and what are the ramifications for modeling and inference?

Periodic sampling introduces sampling correlations, since following selection of a given packet, none of the $N - 1$ following packets are selected. Although this biases against selection of multiple closely spaced packets, there may not be a large impact when sampling from high speed links that carry many flows concurrently. In this case, successive packets of a given flow can be interspersed by many packets from other flows, effectively randomizing the selection. While such randomization may not be effective at lower speed routers carrying fewer flows (e.g. edge routers), packet sampling is not expected to be so necessary for flow formation in this case. In Section 3 we test these expectations by implementing independent and periodic sampling algorithms on packet level traces.

2.4 General Flows and TCP flows

Under independent sampling of packets with probability p , the number of packets k sampled from an original flow of ℓ packets follows the binomial distribution $B_p(\ell, k) = \binom{\ell}{k} p^k (1-p)^{\ell-k}$. In many implementations $p = 1/N$ where N is an integer. In this paper we will assume this to be the case, although the conclusion, and usually the proofs, hold independently of this assumption.

For TCP flows, additional information is available in the flow statistics, at least in NetFlow statistics. The TCP protocol signals the start and end of connections with packets that are distinguished by flags in the TCP header; see e.g. [6]. The first packet of a connection has a SYN flag set; the last has the FIN flag set. A NetFlow statistic includes the cumulative OR of the code bits of flow’s packets. By inspecting the code bits of the flow, we may determine whether a given flag was set in any packet of the flow. We will refer to a packet with a SYN flag set as a SYN packet, and a flow containing a SYN packet as a SYN flow. Here we assume:

- original TCP flows are well-behaved in the sense that they contain exactly one SYN packet.

Under this assumption, the probability that a sampled SYN flow contains a SYN packet is p , the average packet sampling probability. In Section 3.3 we will find that NetFlow traces support the assumption that TCP flows include at least one SYN packet. In packet traces, an overwhelming majority of TCP flows that contained at least one SYN packet, contained exactly one.

In Section 4 we show that the numbers of measured SYN flows of can be used to estimate the number of original TCP flows that were not sampled at all, and hence the total number of original TCP flows. Exploiting the information in the distribution of short sampled SYN flows is essential to making accurate prediction of the distribution of short original flows for scaling-based inference. Although the method applies only to TCP traffic, this is the majority of Internet traffic. In one of the traces used in this study, FLOW, TCP traffic comprises 76% of the flows, 84% of the packets, and 95% of the bytes. Furthermore, the TCP-specific scaling-based method does offer some advantages of the EM-based method in estimating the total number of flows. The relative advantages of the two methods are discussed in Section 9.

A parallel methodology could be based on FIN flags, since all TCP sessions should end with a FIN packet. However, there may be many flows for which this is not the case: a SYN-flooding denial of service attack that employs flows comprising one SYN packet.

2.5 Sparse Flows and Splitting

Packet sampling can actually *increase* the number of measured flows in some circumstances. Given a sampling period N and a flow interpacket timeout T , we say that a given original flow of packets is *sparse* if the typical time between sampled packets ex-

ceeds T . In this case, a single original flow may give rise to multiple flow statistics. Consider an original flow comprising n packets distributed over an interval of duration t . The typical time between sampled packets is tN/n , thus sparseness requires that $tN/(nT) > 1$. It also requires that there is typically more than one sampled packet, i.e., $n/N > 1$. Combining, we can say that the threshold for sparseness is crossed when

$$t/T > n/N > 1. \quad (1)$$

Sparseness is most likely to arise in flows containing many packets occurring with relatively low frequency. It is found that streaming and multimedia applications can generate sparse flows for settings of the sampling parameters within a likely operating range: sampling period $N = 100$ and flow interpacket timeout $T = 30s$; [9].

The potential for flow splitting has ramifications for the present work. Suppose a sparse original flow is split by sampling into a number of sampled measured subflows. With no additional information other than the flow lengths, the best we can hope is to infer the distribution of the lengths of a notional set of original subflows, whose combined length is that of the original flow. Thus in the presence of splitting, we will tend to infer more, and shorter, flows than were actually present. There are three ways to ameliorate this:

Suppression of splitting: increasing the flow interpacket timeout suppresses splitting; from (1) flows are less sparse. However, this remedy has the potential side effect of combining sampled measured flows with the same key that came from distinct original flows. Also, buffer requirements for the flow cache are increased. In general, it may be desirable to systematically change the flow interpacket timeout according to the sampling rate.

Surgery on flows: if there is no control over the flow timeout, another possibility is to emulate the effects of increasing timeout by joining flows with matching keys, qualified by semantic information provided in the flow records. For example, a measured TCP flow containing a SYN packet signifies a starting TCP connection, so should not be joined to a preceding flow with matching key.

Exploiting protocol information: Since the SYN packet at the start of a TCP flow is sampled with probability p , the number of original TCP flows can be estimated from the number of sampled SYN flows. This limits the amount of surgery that should be performed.

2.6 Experimental Packet and Flow Data

The experimental portion of this work was performed using four packet traces and a flow trace. Trace PEERING was derived from 10,000,000 IP packets seen at a peering link during a period of 37 minutes. Trace CAMPUS was derived from 10,065,600 IP packets seen at LAN near the border of a campus network during a period of 300 minutes. Trace ABILENE was collected from an OC48c link in the Abilene network. This study used 532,567,007 UDP and TCP packets present during a 2 hour period in the westbound direction of the Abilene-IPLS-CLEV trace. Trace COS was collected at an OC3 link at Colorado State University. This study used approximately 37 million packets collected during January 25 and 26, 2003; this period was chosen to overlap the onset of the Slammer worm [17], to support the work of Section 8. Further details on ABILENE and COS can be found at [18]. Trace FLOW comprised unsampled raw NetFlow statistics collected in an aggregation network during 1 day in September 2002. There were 229,448,460 records, representing 6,009,481,415 packets and 3,107,927,460,309 bytes.

The packet traces were used as input to applications which sampled packets (either independently or periodically) and formed flow statistics from the sampled stream. The flow key comprised source and destination IP addresses and TCP/UDP port numbers. The flow

packets	Sampling Period N		
	10	100	1000
37M	2×10^{-5}	0.015	0.002
3.7M	0	0.044	0.16
0.37M	0	0.34	0.10

Table 1: Comparing Random and Periodic Sampling: Chi-square P-values, for sampling period $N = 10, 100$ and 1,000, using subportions of trace COS

interpacket timeout was 30 seconds. Protocol specific information, such as TCP SYN or FIN packets, is not used to demarcate flows.

3. EVALUATING THE SAMPLING MODEL

We investigate the dependence of the frequencies of sampled flow lengths on whether periodic or simple random packet sampling is employed. We can formulate this question at two levels. First, we can ask whether the distributions obtained by different sampling methods are statistically distinguishable; standard statistical tests can be used to determine this. But even if two distributions are distinguishable, they may not differ to an extent that concerns us in practice. Our second approach is to formulate a notion of how two distributions might be “close enough” for the purposes of applications using the distributions, and apply it to the measured sampled flow length distributions. In both cases we also investigate the dependence of distinguishability on data size.

3.1 Distinguishing Distributions

Consider two set of sampled flow length frequencies $g = \{g_i : i = 1, \dots, n\}$ and $g' = \{g'_i : i = 1, \dots, n'\}$ created from a set of original flows, g being produced with independent random sampling, and g' by periodic sampling. We take g as our reference distribution, and ask whether g' would be judged as arising from the same distribution. The appropriate chi-squared statistic is:

$$\chi = \sum_i \frac{(g'_i - g_i)^2}{g_i}. \quad (2)$$

We represent it through the associated one-sided chi-squared P-value $P(\chi)$, i.e., the proportion of the time that a value of χ or greater would be obtained if g and g' were drawn from the same distribution. In hypothesis testing we would fix a significance level (probability) P_0 (e.g. 5%) and reject the (null) hypothesis—that g' is drawn from the same distribution as g —if $P(\chi) < P_0$.

Table 1 shows $P(\chi)$ for trace COS and subsets comprising the first $1/10^{\text{th}}$ and $1/100^{\text{th}}$ of the packets. Following the recommendation of §4.3 in [23], we binned adjacent frequencies so as to be no less than 5. The number of bins was 11 in one case, at least 71 in all others. Using a common significance level $P_0 = 5\%$ we see that in many cases the two distributions are *statistically distinguishable*, except when $N = 1,000$ or the smallest fraction of the trace is used. This indicates persistent differences between the two distributions that are not washed out by averaging over long traces. The closer agreement for large N reflects that for higher sampling periods the difference in the sampling algorithms tends to blur because most flows that are sampled have only one packet sampled.

3.2 Distributional Discrepancies

Although the length frequency distributions g and g' obtained by random and periodic sampling can be distinguished, the differences are, in fact, small. Figure 1 displays the frequencies distribution for

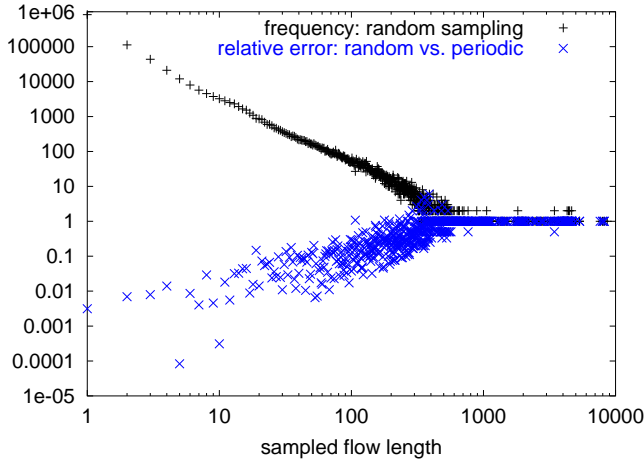


Figure 1: RANDOM SAMPLING VS. PERIODIC SAMPLING. Length frequencies and relative error, COS dataset, $N = 10$.

random sampling, and the relative error $|1 - g'_i/g_i|$. (A small number of points with small $g'_i \geq g_i = 0$ are excluded). Although the relative errors for the larger frequencies are small (1 in 100 or smaller), they remain large enough to distinguish the distributions even for large data sets. Such behavior will occur if, for example, a small subset of the flows are consistently treated differently in the two sampling methods. Although we do not investigate the origin of these discrepancies, a candidate subset is those flow containing packets that are back-to-back in the original stream: successive back to back packets can never both be periodically sampled.

It is desirable to capture the distributional discrepancies in a single measure. Standard measures based on hypothesis tests (such as those used in a related context in [19]) will blow up for large datasets since even small persistent errors will eventually exceed the likely statistical error.

We can deem the two distributions “close enough” for practical purposes, if the typical relative difference between the frequencies is sufficiently small. For a given length i , we normalize the absolute difference between the frequencies by their mean value to obtain the relative difference $2|g_i - g'_i|/(g_i + g'_i)$. To obtain the typical relative difference over all i we average the relative differences that weights them by the mean values $(g_i + g'_i)/2$. Thus we attach more weight to a relative difference of a given size when it occurs for a larger frequency. Altogether, this resulted in the following weighted mean relative difference (WMRD):

$$\text{WMRD} = \frac{\sum_i |g_i - g'_i|}{\sum_i (g_i + g'_i)/2}. \quad (3)$$

The WMRD for trace COS is show in Table 2. For the full trace the WMRD is less than 1% for all sampling periods considered. Similar

packets	Sampling Period N		
	10	100	1000
37M	0.0069	0.0063	0.0015
3.7M	0.023	0.022	0.032
0.37M	0.032	0.039	0.13

Table 2: Comparing Random and Periodic Sampling: WMRD. for sampling period $N = 10, 100$ and $1,000$, using subportions of trace COS

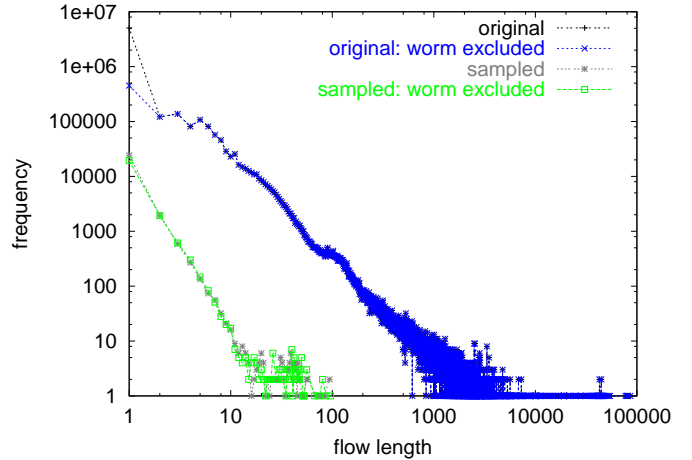


Figure 2: ORIGINAL AND SAMPLED FLOW LENGTH DISTRIBUTIONS: impact of worm packets, sampling rate $N = 1,000$

accuracy was found the traces ABILENE, CAMPUS and PEERING. Although there is no generally agreed standard for the necessary accuracy of length distributions, we expect the accuracy to within 1% will be sufficient for many applications.

3.3 TCP Specific Assumptions

We now examine the assumption of Section 2.4 that TCP flows contain one SYN packet. First: how frequently do TCP flows contain at least one SYN packet? The answer depends on the flow definition: a short interpacket timeout may split a TCP connection into several flows, not all of which contain a SYN packet. We find some indication in the FLOW dataset, where we found that 84% of TCP flows were SYN flows; similar proportions were found in other flow traces. Investigating the same question using packet traces is problematic, since initial SYN packets will not be measured from flows already in progress when trace collection starts.

Second: how frequently do TCP flows contain at most one packet? In the packet traces we determined the proportion of those TCP flows containing at least one SYN packet, that contained exactly one SYN packet. For CAMPUS, it was 98.8%; in PEERING 94.6%.

A single SYN packet in an original TCP flow is expected to be the first packet of the flow. However, our method is insensitive to the position of the SYN packet in the flow, since the probability to sample a single packet is assumed independent of its position.

3.4 Sampling, Components and Nonadditivity

As discussed further in Section 8, trace COS contains packets generated by hosts infected by the Slammer worm. These are manifested as (overwhelmingly) single packet flows. The impact on the original flow length distribution can be seen in Figure 2, which shows the flow length distribution of the original traffic, complete, and with worm packets excluded (upper two curves). The two curves are practically identical except for flows of length 1.

The lower two curves in Figure 2 show the sampled flow length frequencies for independent sampling with period 1,000. The contribution of the work traffic is barely visible at the scale of the aggregate. This reiterates the example of Section 1.1 where we saw that sampled flow length distributions alone does not contain sufficient information to infer all details of the original distribution.

This motivates making maximal use of the information that is present in the flows, or otherwise available. Since different applications are not expected to exhibit the same flow length distri-

butions, traffic may be segmented into classes of interest, e.g. by TCP/UDP port number. Both inference techniques presented in this paper (scaling with enhanced smoothing, and th EM-method) generally produced estimates that are non-additive in the sampled frequencies. Hence segmentation will reduced cross-contamination of estimates between applications, and potentially increase accuracy. As an example, we identify worm traffic by a combination of port number and packet length, and analyze it separately in Section 8.

Finally, Figure 2 illustrates another point. It is sometimes thought that the original flow length distribution can be recovered exclusively by scaling and extrapolating the sampled flow length distribution. This would amount to translating the sampled frequencies curve within the figure. This example shows that no such translation would yield a convincing overlap with the curve of the original frequencies.

4. SCALING-BASED INFERENCE & TCP

Our starting point the simple scaling model described in the introduction: an original flow of length Nk is attributed to each sampled flow of length k . To overcome the limitations described in the introduction, we apply a smoothing to this simple distribution, and use reported TCP flags to draw inferences about flows for which no packet was sampled. Thus this method is limited to inferring characteristics of TCP flows, and assumes the TCP flags are reported in the flow statistics, in the manner of NetFlow.

For a single flow the scaling idea can be made rigorous through Maximum Likelihood (ML) estimation. The idea of ML-estimation is that we are given a family of statistical models each member of which is specified by a parameter value θ . The actual parameter value is to be estimated from measured data X . For each possible parameter θ , the model specifies the probability $P_\theta[X]$ that the measured data would be obtained. The maximum likelihood estimator (MLE) $\hat{\theta}$ is the value which maximizes this probability: $\hat{\theta} = \arg \max P_\theta[X]$. Maximum likelihood estimators enjoy some useful general properties: they are consistent (they converge to the true value as the amount of data grows) and they are efficient (they have minimal asymptotic variance in the same limit).

4.1 Inference of a Single Flow Length

Sampling the packets of an individual flow of length ℓ with probability p should yield a flow of average length ℓp , ignoring splitting. This suggests an inversion based on dividing measured flow lengths by p . The following lemma partially justifies this approach.

Lemma 1. Consider ℓ objects, sampled independently with probability $p < 1$, resulting in k objects. Given k , the likelihood $B_p(\ell, k)$ is maximized at $\ell = \lfloor k/p \rfloor$, unless k/p is an positive integer, in which case $k/p - 1$ has equal likelihood.

When $p = 1/N$ for integral N , both Nk and $Nk - 1$ are equally likely estimators of ℓ . In practice we will select $\ell = Nk$, for the reason that number of packets sampled from an inferred flow is k , on average, i.e., the length of the measured flow.

4.2 Estimation of Length Distribution

We assume a set of n original flows each containing exactly one SYN packet, with f_i the frequencies of flows with i packets. Packet sampling is independent with probability $p = 1/N$. Let g_i be the frequency of sampled flows with i packets. Section 4.1 suggests that we estimate frequencies of the lengths of original flows by ascribing the weight g_j to the original flow length Nj . This approach has the drawback of omitting the mass of flows which had no packet sampled: the total number of flows will be underestimated. Below we describe two methods to estimate the number of unsampled

flows by employing the frequencies $g_i^{\text{SYN}} \leq g_i$ of sampled SYN flows, i.e. sampled flows that contain a SYN packet. Both methods build on the simple scaling method described in the introduction.

4.3 Mean Flow Lengths

As reported in [9], the g_i and g_i^{SYN} provide two different ways to estimate the mean flow length.

Lemma 2. (i) $M^{(1)} = N \sum_{i>1} g_i^{\text{SYN}}$ is an unbiased estimator of the total number of SYN flows.

(ii) $g_0 = (N - 1)g_1^{\text{SYN}}$ is an unbiased estimator of the total number of unsampled SYN flows, and hence, assuming no flow splitting, $M^{(2)} = \sum_{i \geq 0} g_i$ is an unbiased estimator of the total number of SYN flows.

(iii) $P = N \sum_{i \geq 0} i g_i$ is an unbiased estimator of the total number of packets in the original flows.

We form estimators $L^{(i)} = P/M^{(i)}$, $i = 1, 2$, of the mean flow length. Each has distinct advantages. $L^{(2)}$ has lower variance, since $M^{(2)}$ counts all flows. $M^{(2)}$ is more susceptible to bias due to flow splitting, since it counts all sampled flows, where $M^{(1)}$ counts at most only one measured flow from each original SYN flow, i.e. that containing the SYN packet, if sampled.

Having formed estimators of the total number of original flows, and of the mean packets per flow, our task is to distribute the weight of this estimate amongst the possible original flow lengths. The scaling-based estimators that we describe in the rest of this section essentially mirror the two approaches of calculating the mean number of flows, and are subject to the same bias and variance effects.

4.4 Scaling Estimate using only SYN Flows

The first scaling-based estimator uses the counts g_i^{SYN} only. Since the SYN packet is sampled with probability $1/N$, we attribute a weight of N original flows to each sampled SYN flow. For a sampled SYN flow of length j , there are $j - 1$ sampled non-SYN packets, and so we attribute the weight to original flows with $N(j - 1)$ non-SYN packets. Thus, we start by attributing a weight $N g_j^{\text{SYN}}$ to original flows of length $\ell_j = 1 + N(j - 1)$. For $j > 1$ we can smooth this weight over an (integer) interval of width N with ℓ_j as close to its center as possible. This can be done while satisfying the conservation law that the average number sampled SYN flows and sampled packets are equal to g_j^{SYN} and j respectively.

The case $j = 1$ is different. This corresponds to original flows comprising a single SYN packet, and so the number of sampled non-SYN packets is zero. The only smoothing of the weight of $N g_1^{\text{SYN}}$ flows that conserves average flow length is that which concentrates all the weight at length 1. But this is undesirable, since it leaves a gap in the estimated distribution for which there is no particular justification in the data.

Clearly we need to extract more information from the g_i^{SYN} in order to better distribute the weight from g_1^{SYN} . Our strategy here is motivated by the expectation that the dominant contributions to g_1^{SYN} and g_2^{SYN} will be from shorter flows. To see this, assume the extreme case that original flows all have length L . Then $E[g_1^{\text{SYN}}] = f_L N^{-1} (1 - 1/N)^{L-1}$ and $E[g_2^{\text{SYN}}] = f_L (L-1) N^{-2} (1 - 1/N)^{L-2}$, and we would have

$$\frac{E[g_1^{\text{SYN}}]}{E[g_2^{\text{SYN}}]} = \frac{N - 1}{L - 1} \quad (4)$$

The point here is that $E[g_1^{\text{SYN}}] > E[g_2^{\text{SYN}}] \Leftrightarrow N > L$. If in our data $g_1^{\text{SYN}} > g_2^{\text{SYN}}$, then the dominant flow lengths are expected to be in the neighborhood of $L = 1 + (N - 1)g_2^{\text{SYN}}/g_1^{\text{SYN}} < N$.

To smooth the dominant weight, we use a more detailed argument. The aim is to jointly smooth the weights of g_1^{SYN} and g_2^{SYN} uniformly over integer intervals $I_1 = [1, t]$ and $I_2 = (t, \lfloor 3N/2 \rfloor]$ respectively, i.e., the weights at points in the two intervals are $h_1(t) = Ng_1^{\text{SYN}}/t$ and $h_2(t) = Ng_2^{\text{SYN}}/(\lfloor 3N/2 \rfloor - t)$ respectively. (Since higher integer multiples of N are chosen to be at or near midpoints of intervals of with N , the upper boundary $\lfloor 3N/2 \rfloor$ of I_2 lies adjacent to the lower boundary of the interval containing the mass from $j = 3$). Our task is to choose t .

Using the smoothings described above for a given choice of t , the expected number of sampled flows with lengths $i = 1$ or 2 that are generated by original flows with lengths in $I_1 \cup I_2$ are

$$G_i(t) = \frac{h_1(t)}{N} \sum_{j=i}^t B_{1/N}(j-1, i-1) + \frac{h_2(t)}{N} \sum_{j=t+1}^{\lfloor 3N/2 \rfloor} B_{1/N}(j-1, i-1) \quad (5)$$

where $B_{1/N}(\ell, k)$ is the binomial probability $\binom{\ell}{k} (1-1/N)^{\ell-k} / N^k$. From the foregoing discussion, we expect the dominant contributions to g_1^{SYN} and g_2^{SYN} to originate in from shorter flows. This motivates us to choose t such that the ratio $G_1(t)/G_2(t)$ is close to $g_1^{\text{SYN}}/g_2^{\text{SYN}}$. Since t is an integer variable, we cannot expect the ratios to be equal for some t . Instead, we look for

$$t^* = \inf \left\{ t \in [1, \lfloor 3N/2 \rfloor] : \frac{G_1(t)}{G_2(t)} \leq \frac{g_1^{\text{SYN}}}{g_2^{\text{SYN}}} \right\} \quad (6)$$

Finally, we also wish to avoid having t too high; otherwise we run the risk of unduly favoring larger original flow lengths in our estimator, without strong evidence that we should. One measure of the accuracy of the inferred distribution is comparison of the average predicted length with the estimator $L^{(1)}$. A detailed argument that we do not reproduce here shows that further restricting t above enables us to bound the mean length according to the inferred distribution to within a small multiple of $L^{(1)}$. In this paper, we restrict t by saying that $h_1(t)$ shall not be lower than $h_2(t)$, or equivalently, $t < t_{\max} = \lfloor 3N/2 \rfloor g_1^{\text{SYN}} / (g_1^{\text{SYN}} + g_2^{\text{SYN}})$. Summarizing, the value of t we choose is $t^{(1)} = \min\{t^*, t_{\max}\}$ and the inferred original frequencies are (setting $i_N(j) = \lfloor N(j - \frac{1}{2}) \rfloor$)

$$\hat{f}_i^{(1)} = \begin{cases} Ng_1^{\text{SYN}}/t^{(1)}, & i \in [1, t^{(1)}] \\ \frac{Ng_2^{\text{SYN}}}{\lfloor 3N/2 \rfloor - t^{(1)}}, & i \in (t^{(1)}, \lfloor 3N/2 \rfloor], \\ g_j^{\text{SYN}}, & i \in \lfloor i_N(j-1), i_N(j) \rfloor, j \geq 3 \end{cases} \quad (7)$$

4.5 Mixed Scaling Estimator for TCP Flows

We briefly describe a second variant of the scaling approach. We use now the *full* flow counts g_i , but mirroring the approach of Lemma 2(ii), we also use the unbiased estimator $g_0 = (N-1)g_1^{\text{SYN}}$ of the number of unsampled flows. Analogously to Section 4.4, for $i > 2$, the weight g_i is distributed uniformly around the original flow length Ni . The lowest two weights g_0 and g_1 are distributed in two regions $[1, t^{(2)}]$ and $(t^{(2)}, \lfloor 3N/2 \rfloor]$ where $t^{(2)}$ is determined from the lowest two weights g_0 and g_1 analogously to (6); we omit the details. We call the resulting inferred frequencies $\hat{f}_i^{(2)}$:

$$\hat{f}_i^{(2)} = \begin{cases} g_0/t^{(2)}, & i \in [1, t^{(2)}] \\ \frac{g_1}{\lfloor 3N/2 \rfloor - t^{(2)}}, & i \in (t^{(2)}, \lfloor 3N/2 \rfloor], \\ g_j/N, & i \in (i_N(j), i_N(j+1)], j \geq 2 \end{cases} \quad (8)$$

A discussion of the relative advantages of (7) and (8) reflects that of Section 4.3. $\hat{f}^{(2)}$ uses more data, and thus should be subject to

smaller variance. However, $\hat{f}^{(1)}$ is less susceptible to the effects of flow splitting, under our assumptions, since it counts at most one measured flow from each TCP flow. We compare the experimental properties of $\hat{f}^{(1)}$ and $\hat{f}^{(2)}$ in Section 7.2.

5. INVERSION AND ITS DEFICIENCIES

Before proceeding to the full ML estimator, we briefly examine an unbiased estimator, then discard it due to high variance. This estimator exploits the fact the expected values of the sampled frequencies g_i are an invertible function of the original frequencies f_i . Here we assume independent packet sampling with probability p , we ignore splitting, and assume that the original flow lengths are bounded above by some m . Under these assumptions, the g_i have expectation $\mathbb{E}[g_j] = \sum_{i=1}^m C_{ji}(m) f_i$, where $C_{ji}(m) = B_p(i, j)$ for $m \geq i \geq j \geq 1$ and 0 otherwise. We can prove that

Lemma 3. $C(m)$ is invertible: $C_{ij}^{-1}(m) = B_p(i, j)(-p)^{-j-i}$ for $m \geq j \geq i \geq 1$ and 0 otherwise.

This suggests estimating f_i from measured g_i as $\hat{f} = C^{-1}(m)g$, taking as m the maximum desired original flow length. However, this estimator is not well-behaved. The alternating parity with j of the components $C_{ij}^{-1}(m)$ makes estimates very sensitive to variations in g ; some of the estimated frequencies may be negative. This is manifested in the growth of the variance with m : it can be shown that $\text{Var} \hat{f}_i$ grows like $p^{-m} = N^m$. Unless the possible flow lengths are small, variance rapidly makes the estimator useless.

6. MAXIMUM LIKELIHOOD ESTIMATION OF FLOW LENGTH DISTRIBUTIONS

While simple to compute, the multiplicative scaling-based estimators $f^{(1)}$ and $f^{(2)}$ have the disadvantage that their coarseness increases on the scale of N . In this section we present a direct MLE of the original flow length frequencies that, with sufficient data, can provide smoothing at all scales. The method has two versions. The first exploits the sampling properties of SYN flows to estimate TCP flow frequencies; the second does not rely on the properties of SYN flows and hence is not restricted to TCP traffic. In what follows we assume that splitting due to sparseness has been suppressed by any of the means described in Section 2.5.

6.1 ML Estimation for TCP Flows

6.1.1 Likelihood Function and Stationary Points

Let there be n original flows, and let ϕ_i denote the probability that an original flow has i packets. All original flows are assumed to contain exactly one SYN packet. We assume independent packet sampling with probability $p = 1/N$. Our aim is to estimate n and $\phi = \{\phi_i\}$, from the frequencies $g^{\text{SYN}} = \{g_i^{\text{SYN}}\}$ of sampled SYN flows of length i . We now derive an expression for log-likelihood $\mathcal{J}(n, \phi)$ to obtain g^{SYN} given n and ϕ .

The probability the original SYN flow gives rise to a sampled SYN flow is p , i.e., the probability that the SYN packet is sampled. Hence the probability to obtain $\gamma^{\text{SYN}} = \sum_i g_i^{\text{SYN}}$ sampled SYN flows in total is $e^{\mathcal{K}(n)} = B_p(n, \gamma^{\text{SYN}})$. Ignoring splitting, the probability the sampled SYN flow has j packets is $\sum_{i \geq 1} \phi_i c_{ij}$ where c_{ij} is the binomial probability $B_p(i-1, j-1)$. Hence $\mathcal{J}(n, \phi) = \mathcal{K}(n) + \mathcal{L}(\phi)$, where

$$\mathcal{L}(\phi) = \sum_{j \geq 1} g_j^{\text{SYN}} \log \sum_{i \geq j} \phi_i c_{ij} \quad (9)$$

\mathcal{K} and \mathcal{L} can be maximized independently over n and ϕ respectively. The maximizer(s) n^* of $\mathcal{K}(n)$ are as described in Lemma 1. Following Lemma 2(i), we estimate n by $M^{(3)} = \gamma p^{-1}$.

We wish to maximize $\mathcal{L}(\phi)$ subject to the constraints $\phi \in \Delta = \{\phi : \phi_i \geq 0, \sum_i \phi_i = 1\}$. Candidates for the MLE are stationary points of \mathcal{L} . Since \log is concave, so is \mathcal{L} and hence \mathcal{L} has a unique stationary point ϕ^* . Differentiating (9) w.r.t. ϕ_i , subject to the constraint $\sum_i \phi_i = 1$, then ϕ^* must be such that the derivative:

$$\frac{\partial \mathcal{L}(\phi)}{\partial \phi_i} = \sum_j \frac{c_{ij} g_j^{\text{SYN}}}{\sum_{k \geq j} \phi_k c_{kj}} \quad (10)$$

is independent of i for ϕ^* . Any ϕ_i^* for which g_j^{SYN} is proportional to $\sum_{i \geq j} \phi_i^* c_{ij}$ this property, and in particular, the (normalized) inversion estimator $\sum_j (c^{-1})_{ij} g_j^{\text{SYN}}$ found from Lemma 3. As discussed in Section 5, ϕ^* is not guaranteed to lie in Δ : some of the ϕ_i^* may be negative. In this case, the MLE must lie in the boundary of Δ , but not be a stationary point of \mathcal{L} .

6.1.2 Expectation Maximization Algorithm

Location of a non-stationary MLE on a boundary by analytical means is generally difficult. We adopt instead a standard iterative approach: the Expectation Maximization (EM) algorithm [7], whose application we now describe.

- (i) *Initialization.* Pick some initial flow length distribution $\phi^{(0)}$, for example, the estimate obtained in Section 4.2.
- (ii) *Expectation.* Let f_{ij}^{SYN} denote the frequencies of original SYN flows from which j packets are sampled, including the SYN packet. Thus $g_j^{\text{SYN}} = \sum_i f_{ij}^{\text{SYN}}$, while $f_i^{\text{SYN}} = \sum_j f_{ij}^{\text{SYN}}$ is the frequency of original SYN flows of i packets whose SYN packet is sampled. Form the complete data likelihood function assuming known f_{ij}^{SYN} :

$$\mathcal{L}_c(\phi) = \sum_{i \geq j \geq 1} f_{ij}^{\text{SYN}} \log \phi_i c_{ij}. \quad (11)$$

Form the expectation $Q(\phi, \phi^{(k)})$ of $\mathcal{L}_c(\phi)$ conditional on the known frequencies g_j^{SYN} , according to a distribution $\phi^{(k)}$:

$$Q(\phi, \phi^{(k)}) = \sum_{i \geq j} \mathbf{E}_{\phi^{(k)}} [f_{ij}^{\text{SYN}} | g^{\text{SYN}}] \log \phi_i c_{ij} \quad (12)$$

- (iii) *Maximization.* Define $\phi^{(k+1)} = \arg \max_{\phi \in \Delta} Q(\phi, \phi^{(k)})$. Differentiating to find the stationary point $\phi^{(k+1)}$ in the interior of Δ :

$$\phi_i^{(k+1)} = \frac{\mathbf{E}_{\phi^{(k)}} [f_i^{\text{SYN}} | g^{\text{SYN}}]}{\gamma^{\text{SYN}}} = \frac{\phi_i^{(k)}}{\gamma^{\text{SYN}}} \sum_{i \geq j \geq 1} \frac{c_{ij} g_j^{\text{SYN}}}{\sum_{l \geq j} \phi_l^{(k)} c_{lj}} \quad (13)$$

The first equality in (13) arises from the Legendre equations in the maximization of $Q(\phi, \phi^{(k)})$ subject to $\phi \in \Delta$. The second equality can be established through direct computation of the conditional probability. $\phi^{(k+1)}$ can be thought of as refining the estimate $\phi^{(k)}$ as the expected proportions of sampled SYN flows under the probability distribution $\phi^{(k)}$, given the measured frequencies g^{SYN} .

- (iv) *Iteration.* Iterate steps (ii) and (iii) until some termination criterion is satisfied, e.g., some metric distance between successive iterates falls below a specified threshold. Let $\hat{\phi}$ denote the termination point. We write our estimate of the absolute frequencies of original flows as $\hat{f}_i^{(3)} = M^{(3)} \hat{\phi}_i$.

6.2 ML Estimation for General Flows

For general flows—e.g. those using the UDP protocol that has no SYN flag or equivalent—we cannot directly estimate the number of original flows from the number of measured flows. This is

because the probability for a flow to be sampled depends on its length, whose distribution is what we are trying to determine! Instead we adopt a two stage approach. The first stage is to estimate the frequencies ϕ'_i of original flows of length i conditional on at least one of its packets being selected. The second stage is to recover the unconditional distribution.

In order to estimate ϕ' , we can reuse the formulation of Section 6.1.2. This involves constructing analogs of the likelihood functions \mathcal{L} and \mathcal{L}_c for the conditional length distribution, and in particular the iteration (13), with the following changes. Replace g^{SYN} by g , γ^{SYN} by γ ; ϕ by ϕ' and c_{ij} by $c'_{ij} = B_p(i, j)/(1 - B_p(i, 0))$, the probability that j packets are sampled from a flow of length i , conditional on $j \geq 1$, i.e., that the flow is sampled. With this modification, the EM iteration yields an estimate $\hat{\phi}'_i$ of ϕ'_i . The unconditional flow length distribution ϕ is related to the conditional distribution ϕ' through $\phi'_i = \phi_i(1 - B_p(i, 0))/\sum_i \phi_i(1 - B_p(i, 0))$ for $i \geq 1$. We estimate the unconditional distribution as

$$\hat{\phi}_i = \frac{\hat{\phi}'_i/(1 - B_p(i, 0))}{\sum_{i \geq 1} \hat{\phi}'_i/(1 - B_p(i, 0))} \quad (14)$$

The frequencies of original flows are estimated as $\hat{f}_i^{(4)} = \gamma \hat{\phi}_i/(1 - B_p(i, 0))$ and the total number of original flows by

$$M^{(4)} = \sum_i \hat{f}_i^{(4)}. \quad (15)$$

6.3 Issues in Implementation and Execution

Computational Complexity. Let i_{\max} denote the maximum original flow length whose frequency is to be estimated. Tabulation of the binomial coefficients for the iteration is $O(i_{\max}^2)$. Let j_{size} denote the number of non-zero sampled flow length frequencies g_j to be employed. Then each EM iteration is $O(i_{\max} j_{\text{size}})$.

Maximum Sampled Flow Length. In the algorithm, all indices j for which there were sampled flows of length j (i.e. for which $g_j > 0$) were included in the iteration. However, the tails of sampled data sets often exhibit lengths j for which there is only one or a handful of sampled flows, and which are isolated in the sense that there are no neighboring lengths j with $g_j > 0$. In some cases there will be many such flows, even though they represent a small proportion of all flows. Computational complexity of the iteration can be reduced by removing all sampled flows above a certain length j_{\max} , instead treating them with the simple scaling method.

For the general flow estimator $\hat{f}_i^{(4)}$, j_{\max} can be chosen as follows. Consider sampling the packets of an original flow of length Nj independently with probability $1/N$. The average length of the sampled flow is j , and the probability that no packet is sampled is $(1 - 1/N)^{Nj} \approx e^{-j}$. Thus if simple scaling is applied to all flows of length greater than j , the likely error in estimating the total number of corresponding original flows is about e^{-j} . Thus for a given target proportionate error ϵ , we can choose any $j_{\max} \geq j(\epsilon) = \lceil \log(1/\epsilon) \rceil$. For example, $j(10\%) = 3$ and $j(1\%) = 5$. On the other hand, there may be reliable sampled frequencies at lengths longer than $j(\epsilon)$, and these should not be excluded from the iteration. We now discuss criteria for reliability.

Criteria for Use of Sampled Frequencies. The examples discussed toward the end of Section 1.1 show that no inference method can be expected to conjure the true distribution out of thin data. At very high sampling rates, the distribution of sampled flow lengths tends to degenerate onto length 1, with relatively small weight at higher lengths. Conversely, substantially different distributions of original flow lengths may be distinguished by only small differences in the

sampled frequencies. For this reason, the frequencies included in the iteration should have some reliability attached to them.

A basic criterion is that they should be distinguishable from 0. We view a small sampled frequency g_j as a variable that arose from a Poisson distribution whose mean is estimated by g_j . We can say that g_j is distinguishable from 0 at significance level ε if the probability that g_j would have been zero under the Poisson distribution is less than ε . Thus, we require $e^{-g_j} < \varepsilon$, in other words, $g_j \geq j(\varepsilon)$.

Combining with the criterion of the previous item, we see that if $g_j < j(\varepsilon)$ for some $j < j(\varepsilon)$, then it is unlikely the sampled data is sufficiently reliable for inference. Using the same Poisson model, we can associate with a sampled frequency g_j a variance $\sqrt{g_j}$, and so the likely relative error in $1/\sqrt{g_j}$.

Maximum Original Flow Length. We use the likelihood from Lemma 1 to estimate the likely range of values of original flow lengths. Given a sampled flow length j , the (normalized) likelihood $N^{-1}B_{1/N}(\ell, j)$ can be thought of as the posterior distribution of the original flow length $\ell \geq j$ with the non-informative (uniform) prior, i.e., with no knowledge of the distribution of original flow lengths assumed. The mean m_j and variance v_j of this distribution are $N(j+1) - 1$ and $(j+1)N(N+1)$ respectively. Given j_{\max} , we take $i_{\max} = m_j + s\sqrt{v_j}$, i.e., some number of standard deviations above the mean inferred original flow length. Suppose we want to make s large enough that the chance the original flow length exceeds i_{\max} is at most ε . Then we should take $\theta(s) = 1 - \varepsilon$ where θ is the cumulative distribution function of the standard normal distribution. But $1 - \theta(s) < e^{-s^2/2}$ and hence $s < \sqrt{j(\varepsilon^2)}$. Thus as a rule of thumb we can take $i_{\max} \approx N(j_{\max} + \sqrt{j_{\max}j(\varepsilon^2)})$.

Termination of the Iteration. The iterated distributions have smoothness inherited from the binomial probability $b_{1/N}(j, i)$. Sharp features in the original distribution can require extended iteration to resolve, running the risk of noisy or oscillatory behavior in the iterates as a function of original flow length. In practice we have found that termination of the iteration around the onset of such oscillations in the interval $[1, Nj_{\max}]$ is effective in capturing features of the original flow length distribution. The final inferred original frequencies can be obtained by taking the restriction of the iteration based estimate to $[1, Nj_{\max}]$ then concatenating with the scaling estimate obtained from the with frequencies g_j with $j > j_{\max}$.

Comparison of Iterative Methods for TCP. Here, either of the estimators $\hat{f}^{(3)}$ and $\hat{f}^{(4)}$ can be used. Being based on measured SYN flows, $\hat{f}^{(3)}$ is expected to provide better estimates of the total number of inferred flows. On the other hand, it makes use of less data, using only the sampled SYN flows; for this reason the frequency estimates are expected to have higher variance for $\hat{f}^{(3)}$, or, equivalently, be useful for smaller sampling periods N than is $\hat{f}^{(4)}$.

7. EVALUATION AND COMPARISON

In this section we apply the estimators derived in the previous section to experimental traffic traces. Inference is performed on flow statistics derived from sampled version of the traces, and compared with the unsampled flow statistics of the original traces. We compare different estimators applied to the same trace. We use the weighted mean relative difference as a measure of estimation accuracy. In most cases the inferred distributions are accurate to within a few percent. We expect this will be sufficiently accurate for many networking applications.

7.1 Data Considerations

In this section we evaluate performance of the estimators of flow length distributions on the trace datasets. We used the packet trace

datasets described before, CAMPUS, PEERING, ABILENE and COS. In experiments evaluating the TCP-specific estimators $\hat{f}^{(1)}$ and $\hat{f}^{(2)}$, subtraces were extracted from PEERING and CAMPUS as follows. Only packets from the set of original TCP flows (as delineated by a key comprising source and destination IP address and port numbers, and a 30 second interpacket timeout) that started with a SYN packets. This was done primarily to eliminate edge effects: since the traces were collected by packet monitors, SYN packets from flows that started before trace collection commenced will not be present in the trace. This is particularly important for the PEERING trace, whose length 37 minutes, is comparable with the duration of some longer flows in the trace. Restricting the original packet data to flows starting with a SYN packet eliminated 56% of TCP in the (shorter) PEERING trace, and 15% in the (longer) CAMPUS trace. Our characterization of this as an edge effect, rather than deviation from expected TCP behavior, is supported by the fact that in the trace FLOW of NetFlow statistics, the proportion of TCP flows not containing a SYN packet was negligible.

Flow splitting was *not* corrected for, so that original flows may give rise to multiple measured flows. We find below that the estimators perform well despite imperfect conformance with the assumptions underlying Sections 4 and 6.

7.2 Scaling-based Estimators: $\hat{f}^{(1)}$ and $\hat{f}^{(2)}$

We evaluated the performance of the scaling-based estimators $\hat{f}^{(1)}$ and $\hat{f}^{(2)}$ on the CAMPUS and PEERING datasets, for a range of sampling periods N . Packet sampling was performed using deterministic sampling of the original packet stream. Some typical outcomes are displayed in Figure 3, which shows inferred and actual flow length frequencies for the CAMPUS dataset. The right figure uses the estimator $\hat{f}^{(1)}$ for sampling periods $N = 10, 30$ and 100 .

Observe that whereas the inferred frequencies are clearly distinguishable from the actual frequencies due to their stepwise nature, the broad features are similar. At short flow lengths, the management of the widths of the first two steps reflects well the distribution of short flow lengths. Without these manipulations, too little weight would have been attached to the shorter flow lengths. Note that the original distribution has a strong peak at length 5; we cannot hope to distinguish the frequencies of lengths shorter than this using only the weights of the first two steps once N grows much larger than 5. Thus our approach to allocating weights evenly over the first two intervals represents a conservative use of the information available.

The left hand plot in Figure 3 compares the two estimators $\hat{f}^{(1)}$ and $\hat{f}^{(2)}$ at sampling period $N = 30$. Observe $\hat{f}^{(1)}$ more closely captures the peak of the actual frequencies at length 5. $\hat{f}^{(2)}$ is clearly more accurate at longer flow lengths, reflecting its smaller variance due to its use of data from all flows, not just the SYN flows used by $\hat{f}^{(1)}$. (This behavior suggests possibly combining the strengths of the two estimators in their best domains).

In order to compare the accuracy of $\hat{f}^{(1)}$ and $\hat{f}^{(2)}$ quantitatively, we calculated the weighted mean relative difference of each inferred frequencies with the actual frequencies. This was done blockwise, in the sense that for each comparison, the frequencies to be compared were aggregated over the piecewise constant blocks of the scaling-based distribution before the WMRD was calculated. This enables us to factor out the smoothing from the comparison, so comparing the effects of the different choice of block boundaries. The WMRD values are shown in Table 3. This shows $\hat{f}^{(2)}$ to be uniformly better than $\hat{f}^{(1)}$ in predicting the block weights. (As remarked above, $\hat{f}^{(1)}$ better captures the peak frequency). The absolute values of the WMRD look quite good. We expect the few percent error for $\hat{f}^{(2)}$ should be acceptable for use in many net-

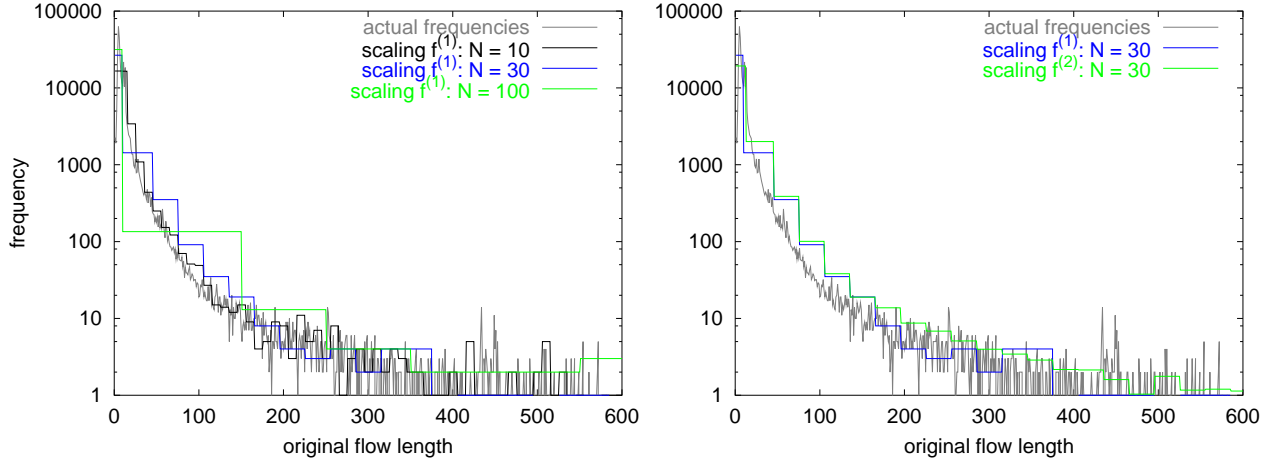


Figure 3: SCALING-BASED ESTIMATORS: original TCP flow length distribution and scaling estimators for CAMPUS dataset. Left: estimator $\hat{f}^{(1)}$ for sampling periods $N = 10, 30$ and 100 . Right: $\hat{f}^{(1)}$ and $\hat{f}^{(2)}$ for $N = 30$. Observe better accuracy for $\hat{f}^{(1)}$ at lower flow lengths, but greater variability at longer flow lengths.

	N		
	10	30	100
$\hat{f}^{(1)}$	8.1%	8.3%	14.5%
$\hat{f}^{(2)}$	4.9%	2.3%	2.7%

Table 3: Weighted mean relative difference of estimated with actual flow length distributions, calculated blockwise. CAMPUS dataset

	N		
	10	30	100
$\hat{f}^{(1)}$	17.2%	20.7%	23.5%
$\hat{f}^{(2)}$	17.9%	18.8%	18.4%

Table 4: Weighted mean relative difference of estimated with actual flow length distributions, unaggregated. CAMPUS dataset

working applications. The WMRD of the unaggregated distributions are shown in Table 4. The WMRD is substantially greater in the unaggregated case, since the smoothing of the inferred distribution takes no account of distribution within blocks. Nonetheless, we expect that discrepancies in the distribution of single flow lengths of roughly 20% may be acceptable for some applications. If only coarser distributional information is required, Table 3 shows the block aggregates to be considerably more accurate.

7.3 ML Estimation with the EM algorithm

7.3.1 Estimation of TCP flow lengths: $\hat{f}^{(3)}$

To evaluate the TCP specific version of the EM estimator, $\hat{f}^{(3)}$, we returned to the CAMPUS and PEERING datasets. Using the termination criteria described in Section 6.3, 5 iterations were performed for sampling period $N = 10$. The WMRD was 5.0% for CAMPUS, noticeably better than the unaggregated WMRD for the scaling based estimators reported in Table 4.

Accuracy was far worse with sampling period $N = 100$, with WMRD about 50%. This is apparently due to insufficient data. For both datasets there are only a total of 100 sampled flows of length greater than 2, and at most 3 flows of any individual length greater than 3, and in both cases there were only 13 flows of length 3.

trace	class	$N = 10$		$N = 100$	
		WMRD	$M^{(4)}$	WMRD	$M^{(4)}$
COS	web	54%	11%	60%	14%
COS	DNS	16%	8%	37%	32%
FLOW	TCP+UDP	-	-	11%	4%
FLOW	DNS/UDP	-	-	3%	3%

Table 5: Weighted mean relative difference in flow length frequencies, and estimation error in total number of flows $\hat{M}^{(4)}$, for COS and FLOW datasets

According to the Poisson analysis of Section 6.3, only sampled frequencies g_j for $j = 1, 2$ have better than around 30% likely accuracy, so it is not surprising that iteration is not very accurate with this amount of data. However, as expected, the TCP based method estimates the total number of TCP flows with $M^{(3)}$ quite well, to within 6% for both $N = 10$ and $N = 100$.

7.3.2 Estimation of general flow lengths: $\hat{f}^{(4)}$

We evaluated the general flow length estimator $\hat{f}^{(4)}$ using the COS trace. We extracted two substraces: web flows (TCP flows with destination port 80) and DNS flows (as identified by destination port 53). The inferred frequencies for $N = 10$ and $N = 100$ are shown in Figure 4. Notice the original web frequencies are not smooth at short lengths; as remarked in Section 6.3 the iterative estimate is smoother than the actual distribution. The DNS flow length frequencies are comparatively smooth: the iterated predictions are fairly close. The $N = 100$ prediction falls off more quickly than the actual frequencies. In this case, only the lowest three sampled frequencies are readily distinguishable from 0, so the lack of accuracy at higher flow lengths is not surprising. For comparison, the total number of sampled flows bears no clear relation to the number of original flows: for DNS flows it was 2% and 20% of the respective original flows for $N = 10$ and $N = 100$; for web, these numbers were 48% and 6% respectively.

The WMRD between the inferred and actual frequencies are shown in Table 5, along with the error in estimating the total number of flows with $M^{(4)}$. Accuracy of inferred frequencies for web traffic is only to within about a factor of 2. This is a result of the ill matching of the smooth inferred distribution to the non-smooth ac-

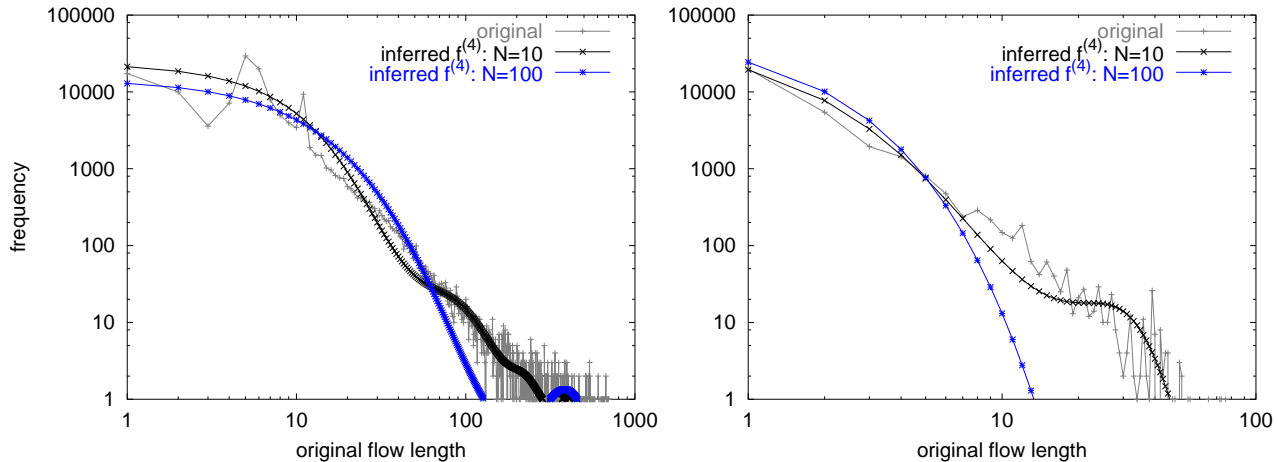


Figure 4: EM-BASED FLOW LENGTH ESTIMATION OF GENERAL FLOWS: Dataset COS: (Left) Web traffic; (Right) DNS traffic. Sampling periods $N = 10$ and 100 . Estimation using $\hat{f}^{(4)}$. Note logarithmic axes.

tual frequencies. Similarly with the scaling-based estimators, there would be closer agreement between slight smoothed (i.e. aggregated) versions of these distributions. Indeed, the total number of flows is estimated to within about 15%, except for the case of DNS with $N = 100$. As mentioned above, there was very little useful sampled data in this case.

Table 5 also show results for sampling period $N = 100$, using two components of the FLOW dataset: all TCP and UDP traffic, and DNS/UDP traffic. The results here were noticeably more accurate than for COS. We believe this is because the original flow length frequencies are somewhat smoother than those of COS. Since FLOW does not contain packet level detail, we emulated the effects of independent packet sampling with probability $1/N$ by taking each flow of length ℓ and generating a random number k of packets following the binomial distribution $B_{1/N}(\ell, k)$. This procedure ignores flow splitting, but using results similar to Theorem 1 of [9], we are able to show that the total number of sampled flows is underestimated by only about 10% on average for sampling period $N = 100$.

8. ESTIMATION OF HOST INFECTIONS

As a concrete example we wish to estimate the number of hosts that have been compromised in a network attack, and are themselves sending out attack traffic. In particular, we wish to estimate the number of such hosts that send traffic past a given collector or set of collectors of sampled flow statistics. By combining with routing information, we may identify the numbers of infected hosts in different regions of the network.

A recent example of such an attack arose in the MS SQL server worm that started activity on January 25, 2003; see e.g. [17]. Infected hosts send out a sequence of attacking packets to randomly chosen destination IP addresses. The source IP address is not spoofed. The attack packets have the following signature: a 404 byte UDP packet with destination port 1434. Since the destination IP address of attack packets is chosen randomly from packet to packet, it is very unlikely that two attack packets with the same destination IP address will be present from the same attacker within the flow interpacket timeout. Indeed, [17] reports the largest directly observed attack rate from a host of 26,000 scans per second. At this rate, the chance that a given 32 bit address will recur within a 30 second pe-

riod is about 0.02%. Thus it is reasonable to assume for simplicity that all attack packets give rise to a single packet flow statistic.

Assume packet sampling with probability $p = 1/N$. Since the original flows comprise one packet, then each attack flow is present in the sampled flow statistics with probability p . Thus, counting the number of distinct source IP addresses will undercount the number of infected hosts: some hosts may have none of their attack packets sampled.

We can map the problem of detecting the number of hosts onto the problem previously solved. For each attacking host (i.e. source IP address matching the profile) represented in the sampled flow statistics, we compute the number i of attack flows detected in the sampled flow statistics. Let g_i denote the absolute frequency of hosts sourcing i measured attack flows. Then we estimate the distribution of the actual number of hosts sourcing i attacks flows using the estimator $\hat{f}^{(4)}$. The total number of infected hosts is estimated as $M^{(4)} = \sum_{i \geq 1} \hat{f}^{(4)}$.

We tested the method on the COS dataset. Within the trace there were 4,542,157 worm packets originating from 49,200 hosts. However, the distribution was highly skewed in the tail: three hosts originated 3,005,083 and 978,841 and 38,770 of the worm packets seen in the trace, i.e. at least 88% of the total worm packets. All other hosts originated less than 2,250 packets each. We conjecture that the hosts generating the largest numbers of packets were located in the campus at which the trace was taken; in this case, most of the randomly chosen target addresses would be on external networks and hence be routed past the trace collector.

We performed inference of the total number of attacking hosts for $N = 10$ and $N = 100$. For $N = 10$, we took $j_{\max} = 50$; there were $M_+ = 72$ hosts originating more than 50 packets. Oscillatory behavior in the inferred distribution onset at about 100 iterations. At this point there were 43,403 inferred hosts; added with M_+ this yielded about 88% of the true number. For $N = 100$, we took $j_{\max} = 10$; there were $M_+ = 16$ originating more than 10 sampled packets. Oscillatory behavior of the inferred distribution onset after about 1000 iterations. At this point there were 27,178 inferred hosts; added with M_+ this yielded about 55% of the true number. By comparison, the number of sampled hosts were 14,667 for $N = 10$, and 3,469 for $N = 100$.

For our problem, the first order unsmoothed jackknife estima-

tor D_{uj1} from [13] takes the form $D_{uj1}(N) = d(N)/(1 - (1 - 1/N)g_1/q(N))$ where $d(N)$ is the number of hosts represented in the sampled trace, and $q(N)$ the number of sampled packets. $D_{uj1}(10) = 14,912$ and $D_{uj1}(100) = 3,672$, little different from the number of *sampled* hosts. Other estimators recommended in [13] are refinements of D_{uj1} , and exhibit roughly the same behavior. To be fair, these estimators are intended for use when the sampling rate is not very small; their representation of the frequency distribution through a limited number of moments cannot be expected to capture the high variability of actual frequencies.

9. CONCLUSIONS AND FURTHER WORK

This paper was motivated by the desire to understand detailed flow statistics of Internet traffic on the basis of flow statistics compiled from sampled packet streams. Increasingly, only sampled flow statistics are available: inference is required to determine the flow characteristics of the original unsampled traffic.

In this paper we have proposed using two inference methods. The scaling method codified the heuristic that when sampling 1 out of N packets, since sampled flows have roughly $1/N$ of their packets sampled, the length of the original flow should be N times the sampled flow. To pin this down we needed to estimate the number of unsampled flows, this required extracting additional information in the form of the number of sampled SYN packets.

In the scaling approach, the hard work was in adjusting the lowest order weights to better reflect the underlying distribution. Clearly there is scope to extend this approach to better tune the distribution of weights from longer flows. An open question is whether a similar analysis for general flows can be used estimate the number of unsampled flows in the absence of protocol information.

The EM algorithm is an iterative approach to ML estimation of flow length frequencies. It does not require protocol level information, although it can exploit it. The versatility comes at the cost of computational complexity, and less control over the total number of inferred flows. The main challenge for the method is selection of a good termination criterion. Prolonged iteration was found to lead to some oscillatory behavior in the tail of the inferred distribution. Our rule of thumb was to terminate before such oscillations become pronounced. Estimation of the head of the distribution was found to be reasonably accurate at this point, to within a factor of 2 at worst, down to a few percent in some cases. In future work we propose to augment the EM approach with second order methods to achieve faster convergence. Another avenue is to use prior statistical models for the distributions to favor conformance with model distributions by use of penalized likelihood [15].

Acknowledgments

Thanks are due to Jia Wang for coding a version of the EM-inference algorithm used in some experiments, and to Balachander Krishnamurthy and Joerg Michael for assistance with some of the datasets.

10. REFERENCES

- [1] J. Apisdorf, K. Claffy, K. Thompson, R. Wilder, "OC3MON: Flexible, Affordable, High Performance Statistics Collection," See: <http://www.nlanr.net/NA/Oc3mon>
- [2] B.-Y. Choi, J. Park, Zh.-L. Zhang, "Adaptive Random Sampling for Load Change Detection", ACM SIGMETRICS 2002 (Extended Abstract).
- [3] Cisco NetFlow; for further information see <http://www.cisco.com/warp/public/732/netflow/index.html>
- [4] 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. 13, no. 8, pp. 1481–1494, Oct. 1995.
- [5] K.C. Claffy, G.C. Polyzos, and H.-W. Braun. "Application of Sampling Methodologies to Network Traffic Characterization", Proceedings ACM SIGCOMM'93, San Francisco, CA, September pp. 13–17, 1993.
- [6] D. Comer, "Internetworking with TCP/IP, Volume 1: Principles, Protocols, and Architecture", Third Edition, Prentice Hall, NJ, 1995.
- [7] A.P. Dempster, N.M. Laird, D.B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm (with discussion)", J. Roy. Statist. Soc. Ser., vol. 39, pp. 1–38, 1977.
- [8] N.G. Duffield, C. Lund, M. Thorup, "Charging from sampled network usage," ACM SIGCOMM Internet Measurement Workshop 2001, San Francisco, CA, November 1-2, 2001.
- [9] N.G. Duffield, C. Lund, M. Thorup, "Properties and Prediction of Flow Statistics from Sampled Packet Streams", ACM SIGCOMM Internet Measurement Workshop 2002, Marseille, France, November 6-8, 2002.
- [10] C. Estan and G. Varghese, "New Directions in Traffic Measurement and Accounting", Proc SIGCOMM 2002, Pittsburgh, PA, August 19–23, 2002.
- [11] A. Feldmann, R. Cáceres, F. Douglass, G. Glass, M. Rabinovich, "Performance of Web Proxy Caching in Heterogeneous Bandwidth Environments," in Proc. IEEE INFOCOM'99, New York, NY, March 23-25, 1999.
- [12] A. Feldmann, J. Rexford, and R. Cáceres, "Efficient Policies for Carrying Web Traffic over Flow-Switched Networks," IEEE/ACM Transactions on Networking, vol. 6, no.6, pp. 673–685, December 1998.
- [13] P.J. Haas and L. Stokes, "Estimating the number of classes in a finite population," J. Amer. Statist. Assoc., vol. 93, pp 1475–1487, 1998.
- [14] Inmon Corporation, "sFlow accuracy and billing", see: <http://www.inmon.com/PDF/sFlowBilling.pdf>
- [15] P.J. Green, "On the use of the EM algorithm for penalized likelihood estimation," J. R. Statist. Soc. B, vol. 52, pp. 443–452, 1990.
- [16] "Internet Protocol Flow Information eXport" (IPFIX). IETF Working Group. See: <http://net.doit.wisc.edu/ipfix/>
- [17] D. Moore, V. Paxson, S. Savage, C. Shannon, S. Staniford, N. Weaver, "The Spread of the Sapphire/Slammer Worm", Technical Report, CAIDA, 2003. See <http://www.caida.org/outreach/papers/2003/sapphire/sapphire.html>.
- [18] NLANR Moat PMA trace archive. See <http://pma.nlanr.net/Traces/long/ipls1.html>
- [19] V. Paxson, "Empirically-Derived Analytic Models of Wide-Area TCP Connections", IEEE/ACM Transactions on Networking, Vol. 2 No. 4, August 1994.
- [20] V. Paxson, G. Almes, J. Mahdavi, M. Mathis, "Framework for IP Performance Metrics", RFC 2330, May 1998.
- [21] Packet Sampling (PSAMP) IETF Working Group Charter. See <http://www.ietf.org/html.charters/psamp-charter.html>
- [22] J. Postel, "Transmission Control Protocol," RFC 793, September 1981.
- [23] L. Sachs, "Applied Statistics", Second Edition, Springer, New York, 1984.
- [24] C.F. Jeff Wu, "On the convergence properties of the EM algorithm", Annals of Statistics, vol. 11, pp. 95–103, 1982