# On Passive One-Way Loss Measurements Using Sampled Flow Statistics

Yu Gu* Lee Breslau† Nick Duffield† Subhabrata Sen†
*NEC Labs, America, Princeton, NJ †AT&T Labs–Research, Florham Park, NJ
Email: yugu@nec-labs.com Email: {breslau,duffield,sen}@research.att.com

***Abstract*—The ability to scalably measure one-way packet loss across different network paths is vital to IP network management. However, the effectiveness of active-measurement techniques depends on being able to deploy measurement hosts at appropriate locations, and to inject necessary amounts of probe traffic without impacting the performance of interest. On the other hand, existing passive-measurement methods like [1] require router support and suffer from deployment limitations for the foreseeable future. In this paper, we propose a new estimation technique that does not require any new router features or measurement infrastructure, and only uses the sampled flow level statistics that are routinely collected in operational networks. The technique is designed to handle challenges of sampled flow-level aggregation such as information aggregation and non-alignment of flow records with measurement intervals. We develop three different schemes and derive analytical bounds on the variance of loss estimation from such a flow-based approach. Our analysis shows that link data rates are now becoming sufficiently large to counteract the effects on sampling on estimation accuracy.**

## I. Introduction

*Motivation*— The ability to accurately and scalably monitor the network's health has become vital to critical network management functions like anomaly detection, troubleshooting, and SLA compliance monitoring. One particularly sought-after capability in IP network measurement/management is that of accurately and scalably measuring the one-way packet loss experienced by traffic along a specific path from some router $A$ to another router $B$ in the network.

The importance of network loss performance monitoring has led to a lot of research efforts on related areas. Most of the existing or proposed approaches fall into one of the following three directions:

*Active Measurement*, which directly measures packet loss rate by exchanging probe traffic between host pairs;
*Network Performance Tomography*, which shares many of the general properties of active measurement, but performance on component links is inferred by correlating measurements on intersecting paths through them; and
*Passive Measurement*, which exploits observations of traffic at two measurement points to infer intervening path performance.

Regarding accuracy and scalability, the first two approaches share some common limitations. First, there is always the argument on whether the measurement results truly reflect the performance of the underlying target traffic. These two approaches obtain the measurement results from collecting the performance of probing packets. However, the probing traffic may have already distorted the underlying traffic and its performance can differ from that of the underlying traffic. Second, the coverage of these two approaches is limited to paths joining deployed measurement hosts and extensive deployment of the measurement hosts brings high deployment and management costs as the network scales. Besides, for the tomography approach, correlated measurement generally requires finer resolution and more complexity in the measurement infrastructure, e.g. the ability for measurement endpoints to report observations on small groups of packets or even individual packets.

On the other hand, existing passive measurement approaches require extra router support and suffer from deployment limitations. For example, trajectory sampling [1] has been proposed as a method to correlate sampling of traffic at different locations. Routers sample packets only if a hash calculated over packet fields falls in a given set; see also [2]. This hash-based selection is being standardized but is not currently available as a standard router feature. In the meanwhile, we are motivated to develop complementary passive measurement techniques that require only the existing deployed network infrastructure to estimate one-way path loss and do not require any new router features or measurement infrastructure.

As an alternative, interface packet drop counts are ubiquitously available from routers via SNMP polling. However, losses not specific to an interface may not be reported. In addition, temporal granularity is limited by the SNMP polling frequency, commonly several minutes, and polling intervals are not expected to be synchronized across routers in a large network. Both these properties make it problematic to compose link measurements along a path in order to estimate path loss, since the link measurement periods are not aligned, and non-interface path loss is missed.

*Contribution*— In this paper we propose correlating flow measurements—as exemplified by NetFlow [3]—of the same network traffic taken at different observation points, in order to infer loss on the intervening path. Our prime example is of inferring link loss rates from flow measurements gathered from routers at either end of a network link. As we shall see, traffic rates on high speed links are only now reaching levels that allow this passive measurement technique to be

This work was done while Yu Gu was a summer intern in AT&T Labs.

practical under the effect of statistical noise due to packet sampling that must typically precede the formation of flow statistics. The technique is particularly attractive because it allows baseline loss measurements to be performed across the network requiring no modification to routers or deployment of other measurement devices. To the best of our knowledge, this is the first proposal to use currently available flow statistics for performance measurement, as opposed to traffic analysis and engineering applications.

A passive measurement approach based on these commonly available flow reports has the following three advantages. First, it is *accurate*. Flow reports are generated from underlying traffic, not the probing traffic. Therefore, the performance extracted from these flow records reveals those of the underlying traffic without any distortion. Second, it is *scalable*. As flow reporting has been widely deployed in commercial routers, obtaining flow reports network wide does not put extra requirement on deployment or new router features. And third, it is *flexible*. Under certain circumstances, one can even extract flows of a particular kind, e.g. TCP flows targeting at port 80 (web traffic), and distinctly analyze their performance.

We develop a simple model that analyzes the effect of sampling on the inference accuracy. We also propose three estimation algorithms that address the problem of aggregation in flow records and coordinating flow records collected at different routers. The analysis demonstrates that given today's Internet traffic, our estimators can discern packet loss rate of $1\%$ or smaller over a one minute measurement interval, which approximates the accuracy of an active measurement at a rate of 2 probing packets per second. Furthermore, as the network evolves and the traffic volume increases, our algorithms will provide estimates with better accuracy.

We have evaluated our approaches over multiple real traces collected at different vantage points in different networks and at different times. By using simulations, we are able to control the true underlying loss process and compare with estimates from our algorithms. The results demonstrate accurate estimations with low variability, close to what the analysis predicts. For instance, we are able to discern a $0.5\%$ packet loss rate over a one minute period under a low packet sampling rate of $1/500$ from a stream of 2 Gb/sec.

## II. Problem Setting and Challenges

We first review how flow records are formed by routers, and the information reported by them. In the flow paradigm, exemplified by NetFlow, routers export flow records that summarize groups of packets with a distinguishing common property, known as the key, observed within a period of time. The distinguishing flow key is commonly built out of the packet header 5-tuple, i.e., protocol, source and destination IP address and TCP/UDP port. Flows are terminated, i.e., the summaries are closed out and exported, when any one of a number of conditions occurs, including (i) inactive timeout (time since a flow's previous packet exceeds a threshold) (ii) active timeout (time since a flow's first packet exceeds a threshold) (iii) protocol events (e.g. TCP FIN flag observed) or (iv) router cache flushing. The flow records report the flow key, total bytes and packets, time of first and last observed packets, cumulative OR of TCP flags over all observed packets. Ingress and egress interfaces of a flow through the router are also reported: this enables us to select, from all flow records produced from a router, those that record traffic traversing a given link.

### A. A Simple Model for Loss Estimation

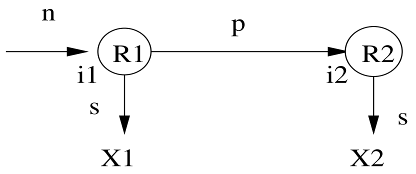


Fig. 1. Model: Routers $R_1$, $R_2$; $n$ packets traverse link $R_1 \rightarrow R_2$ via interfaces $i_1$ and $i_2$, via packet sampling rate $s$; numbers of sampled packets $X_1, X_2$ and packet loss rate $p$.

Let $\mathcal{P}$ be a path in the network and $R_1$ and $R_2$ be two routers in the path. Packets going through path $\mathcal{P}$ will first arrive at interface $i_1$ on $R_1$ and then arrive at interface $i_2$ on $R_2$. Between $i_1$ and $i_2$, packets are subject to a loss with rate $p$. We assume that $i_1$ and $i_2$ can export flow statistics compiled from a substream of packets that has been independently sampled at each router with the same sampling rate $s$. Our analysis below extends simply to the case of unequal sampling rates. Our problem is, *During a specified time interval,* $(t_1, t_2)$*, how can we use the flow level statistics to infer* $p$*?*

Consider first an ideal model in which packets are lost with the same probability $p$, not necessarily independently. We are provided with counts $X_1$ and $X_2$ of all packets sampled at $i_1$ and $i_2$ from the set of $n$ packets that traverse both $i_1$ and $i_2$ during an interval $(t_1, t_2)$; see Fig. 1. $X_j/s$, $j = 1, 2$, are unbiased estimators of the actual number of packets in question traversing $i_j$, and hence we can estimate $p$ by

$$\hat{p} = 1 - \frac{X_2/s}{X_1/s} = 1 - \frac{X_2}{X_1} \qquad (1)$$

Under some very general conditions on the loss and sampling processes on the stream of packets, $\hat{p}$ converges almost surely to $p$ as the number of underlying packets $n$ grows. The precise condition is that loss and sampling are ergodic processes, a class that includes independent (i.e., Bernoulli) and certain correlated processes (e.g. Markovian) as examples.

In this paper we first consider a simple Bernoulli model for loss and sampling, primarily because the variance of the loss rate estimator can be computed exactly. We then consider how much estimator behavior differs from its predictions under two departures from the model: time dependence and loss correlation. When loss rates are time varying, the estimator reflects average loss over the measurement interval. Correlated losses will increase estimator variance to some degree, but we expect the impact to be small if congestion timescales are substantially smaller than the measurement interval. To address this, in Section IV, we explicitly model temporal loss correlations using a Markovian model, and find that the estimator works well even under these conditions.

### B. Challenges

The ideal model appears accurate when flows report single packets, e.g., with zero inactive timeout, for then the (first) packet timestamp reported in the flow can be used to locate a reported packet within the measurement interval. However, the wide use of non-zero timeouts generally prevents us from exactly locating sampled packets with a precise timestamp. This manifests itself in the following two ways:

*Information Aggregation.* Since only the time of first and last sampled packet are reported, if a flow reporting more than two sampled packets intersects with the measurement interval but is not completely contained within it, we cannot say with certainty how many of the flow's packets were sampled during the measurement interval.

*Nonalignment of Flow Records.* At a given router, a single underlying set of packets with a common key can lead to the generation of multiple NetFlow records and the flow start time and flow end time recorded in these records are decided by packet sampling as well as active timeout, inactive timeout and flow cache full events. Since each router generates NetFlow records independently, these events will happen at different times for the same flow at different routers. As a result, NetFlow records generated by the same flow at two different routers may not align themselves in time (see Fig. 2). These unaligned NetFlow records further decrease the ability to attribute sampled packets to a measurement interval. For example, a given packet may be reported in a flow record from $R_1$ that lies entirely within the measurement interval, but in a record from $R_2$ that does not.

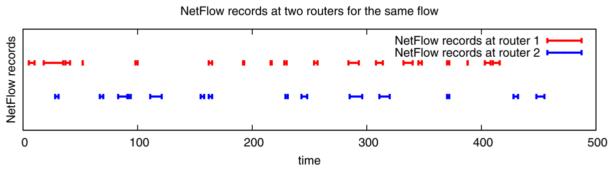


Fig. 2. NetFlow records are unaligned in time

## III. Loss Estimation Algorithms

In this section, we first present three approaches to address the challenges arising from the flow paradigm. Then we analyze the variance brought in from the sampling effect.

### A. SYN/FIN based method

SYN or SYNACK flags are set in the first packets of a TCP session. Therefore, if a SYN is sampled, we can assume that the flow start time in the NetFlow record is the arrival time of the SYN packet. Similarly, as the FIN packet is the packet declaring the end of a TCP session, we can assume that the flow end time in the NetFlow record is the arrival time of the FIN packet. As a result, if the SYN or FIN packet is sampled by NetFlow, we know its arrival time from the flow start time or flow end time reported in the NetFlow record. If this time is within the measurement time interval $(t_1, t_2)$, we can then include it in the packet count $X_i$ for the router $i$ that generated the record. This effective reduces the analysis to the ideal case and we use the loss rate estimator (1).

### B. Fitted flows based method

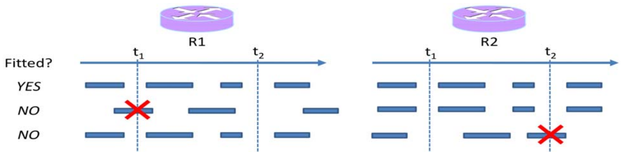


Fig. 3. Flows that are 'fitted'

The key idea of SYN/FIN based method was to unambiguously localize SYN and FIN packet arrival times within the measurement interval. The fitted flows based (Fitted) method extends this idea to a larger set of packets. We inspect NetFlow records generated at both routers for all the flows. For a flow $F$, let $\mathcal{R}_F^1$ be the set of NetFlow records generated by $F$ at router $R_1$ and $\mathcal{R}_F^2$ be those generated at router $R_2$. We call a flow *fitted* if for all NetFlow records $R \in \mathcal{R}_F^1 \cup \mathcal{R}_F^2$, the starting time $t_s$ and end time $t_e$ recorded in $R$ either fall strictly within the measurement interval $t_1 \leq t_s < t_e \leq t_2$, or fall completely outside the interval $t_s < t_e < t_1$ or $t_2 < t_s < t_e$. See Fig (3). For these fitted flows, we can obtain an unbiased estimator of the total number of packets from their flow records arriving during the measurement interval. In this method, we also add in the number of SYN and FIN packets recorded during the measurement interval from the unfitted flows.

### C. Weighted flows based method

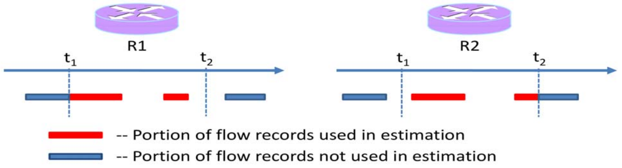

R1
R2
t1
t2
t1
t2
-- Portion of flow records used in estimation
-- Portion of flow records not used in estimation

Fig. 4. Flow statistics using Weighted method

The reason that we do not use the NetFlow records for those 'unfitted' flows in the Fitted flows method is that for those NetFlow records not entirely within the measurement interval, they contain both packets arriving during the measurement interval and packets arriving before or after the measurement interval. This aggregation makes it difficult to determine the number of packets that actually arrive during the measurement interval. However, if we assume that the rate of sampled packets within the flow is relatively smooth, we can then try to utilize these NetFlow records by taking part of the sampled packets in proportion to the duration overlapped with the measurement interval. The smoothness assumption is reasonable when connection round trip times (that govern burstiness) are far smaller than the flow duration. Specifically, for an expected range of packet rates and sampling parameters one can argue that less than one packet will be sampled from each TCP flight, and hence there will typically be no bursts of sampled packets.

Let $t_R^s$ denote the flow start time recorded in NetFlow record $R$ and $t_R^e$ denote the flow end time. The duration of the NetFlow record $d_R$ is then $d_R = t_R^e - t_R^s$. And let the duration of the overlap of $(t_R^s, t_R^e)$ and $(t_1, t_2)$ be $o_R$, then we have, in this method

$$X_1 = \sum_{R \in \mathcal{R}_1} \frac{o_R}{d_R} n_R, \quad X_2 = \sum_{R \in \mathcal{R}_2} \frac{o_R}{d_R} n_R$$

where $\mathcal{R}_i$ is the NetFlow records generated at router $R_i$ and $n_R$ is the number of packets reported in NetFlow record $R$ (see Fig 4.) We then use the loss rate estimator (1) as before.

### D. Analysis of Estimator Variance

We have seen in Section II that $\hat{p}$ is a consistent estimator of the loss rate $p$. Under the Bernoulli loss model, the speed of convergence of the numerator and denominator of (1) can be determined using the Central Limit Theorem, and the Delta-method [4] then enables us to approximate the variance of $\hat{p}$ as the number of underlying packets $n$ becomes large. Specifically, the Delta method lets us approximate the variance of $\hat{p}$ as $\mathsf{Var}(\hat{p}) \approx \nabla f \cdot C \nabla f$ where $\nabla f$ is the vector of partial derivatives of $f$ evaluated at the expected values $\mathsf{E}[X_i]$, and $C$ is the covariance matrix of $(X_1, X_2)$. $X_1$ and $X_2$ are independent and one finds

$$\mathsf{Var}(\hat{p}) \approx \frac{1}{ns} \left( 2(1-p)^2(1-s) + p(1-p) \right) \qquad (2)$$

Note that for small loss and sampling rates $s$ and $p$ we have $\mathsf{Var}(\hat{p}) \approx 2/(ns)$. Note that this is independent of the underlying loss rate $p$.

How many underlying packets $n$ are required in order to reach certain loss estimation accuracy under different sampling rates? A fundamental requirement is that we must be able to *discern* the loss rate, in the sense that statistical fluctuations of its estimator must not be so large that the estimated loss rate will often be zero. A analytic way to express this is to require that the standard deviation (SD) of the estimated loss be smaller than the actual loss rate, i.e., $\sqrt{\mathsf{Var}(\hat{p})} \leq p$. Equivalently, given an average data rate of $r$ packets per second, we require a measurement interval of duration $\tau \geq 2/(srp^2)$. Fig. 5 shows the log of the SD that can be achieved with $n$ packets under a sampling rate of $s$ when the packet loss rate is $p = 0.01$. For example, suppose the packet sampling rate is $s = 1/500$, the figure indicates that in order to have a SD that is less than $p = 0.01$, $n$ should be approximately $10^7$.

To compare with active measurement of the same loss, consider $n$ packets subject to independent loss at rate $p$, with $X$ packets surviving. The resulting estimate of $p$ is $\tilde{p} = 1 - X/n$ with variance $\mathsf{Var}(\tilde{p}) = p(1-p)/n \approx p/n$. Thus for a given number of packets traversing the network $\frac{\mathsf{Var}(\tilde{p})}{\mathsf{Var}(\hat{p})} \approx sp/2$ In the example $s = 1/500$, $p = 0.01$, this ratio is $10^5$, i.e, active measurement needs a factor $10^5$ fewer packets to achieve the same accuracy. Thus collecting $10^7/10^5 = 10^2$ packets in the same time frame of 1 minute requires a probing rate of $100/60 \approx 2$ packets per second.

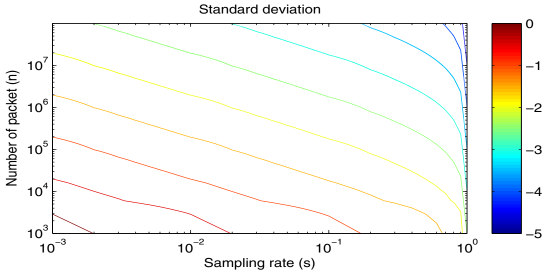


Fig. 5. Log of SD under different sampling rate and sample size

## IV. Evaluations

In this section, we present evaluation results for the above three techniques. We want to understand the accuracy that these estimators can achieve under different circumstances, in particular, under Markov modulated loss process, which has been used to model real Internet loss processes which contains correlation and burstiness [5], and under multiple measurement intervals, active and inactive timeout regimes, sampling rates, and traffic data rates etc.

### A. Methodology and Setting

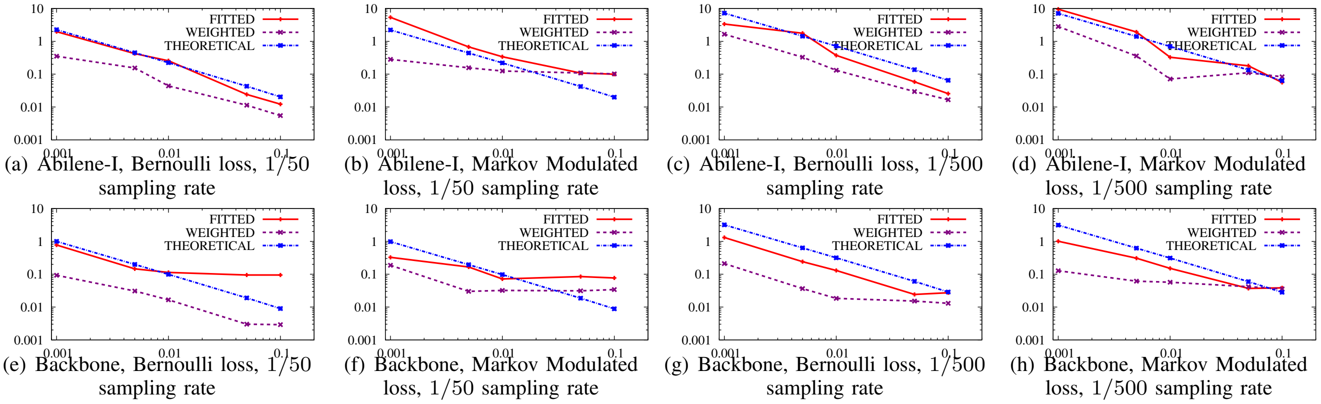
The evaluations used 14 real world traffic traces from different vantage points collected over several years. We observe similar result trends for all these traces, and in the interest of space, only present results from two typical traces collected between 2002 and 2008, at different geographic locations: a trace from the Abilene network available in the NLANR Special Traces Archive [6]; and a backbone link trace of a Tier-1 ISP. Table I summarizes some of the characteristics of the traces.

For each trace, we generate another 10 traces with different loss rates or processes. The losses are generated either using a Bernoulli loss process or a Markov modulated loss process. The average loss rates are set to 0.001, 0.005, 0.01, 0.05 and 0.1 respectively. For the Markov modulated process, we define two loss states with different loss rates. The high loss rate is 10 times the low loss rate and the average sojourn time in the high loss state is 0.5 second while that in the low loss state is 10 seconds. The original trace serves as the input traffic to the first router and the traces with loss serve as the input traffic to the second router on the network path being measured.

We built a NetFlow simulator that generates packet sampled NetFlow records. The sampling rate is set to be 1 (no sampling), 1/50 and 1/500 respectively. The simulator also considers different active and inactive timeout thresholds. We considered three (active timeout, inactive timeout) combinations: $T1$: (60$s$,15$s$), $T2$: (1800$s$,60$s$) and $T3$: (300$s$,30$s$). These represent short, long and medium-sized timeout regimes and the values are based on those used in real networks. We also evaluate our estimation techniques for measurement intervals of 30$s$, 60$s$, 300$s$ and 900$s$, respectively.

| Trace | Date | Link Speed | Duration | Data rate | Packet rate | SYN/FIN rate |
|---|---|---|---|---|---|---|
| Abilene-I | 2002, Aug, 14th | OC-48(2.5Gbps) | 20 min | 400 Mbps | 64136 pkts/s | 1015 pkts/s |
| Backbone | 2008, Jun, 17th | OC-192(10Gbps) | 30 min | 2 Gbps | 327416 pkts/s | 25891 pkts/s |

TABLE I
TRACES USED IN EVALUATION



(a) Abilene-I, Bernoulli loss, $1/50$ sampling rate

(b) Abilene-I, Markov Modulated loss, $1/50$ sampling rate

(c) Abilene-I, Bernoulli loss, $1/500$ sampling rate

(d) Abilene-I, Markov Modulated loss, $1/500$ sampling rate

(e) Backbone, Bernoulli loss, $1/50$ sampling rate

(f) Backbone, Markov Modulated loss, $1/50$ sampling rate

(g) Backbone, Bernoulli loss, $1/500$ sampling rate

(h) Backbone, Markov Modulated loss, $1/500$ sampling rate

Fig. 6. Relative errors for two traces with a high data rate and a low data rate respectively

### B. *Evaluation results*

We calculate the mean relative error from relative errors over all measurement intervals of the whole trace as a measure of the estimation accuracy

$$\text{Mean Relative Error} = \Big(\sum \frac{|\hat{p}_i - p|}{p}\Big)/k.$$

where $\{\hat{p}_1, \ldots, \hat{p}_k\}$ are $k$ estimation results and $p$ is the underlying true packet loss rate. Figure 6 demonstrates the mean relative error obtained under different packet loss rates for traces $Abilene - I$ and $Backbone$, which have low and high data rates. The results are obtained using flow records generated under timeout scheme $T1$ and the measurement interval is set to 300 seconds. For each trace, four sets of results corresponding to different loss processes and sampling rates are presented. Here, we also calculated the theoretical relative error values $\sqrt{\mathsf{Var}(\hat{p})}/p$ as a reference. We see that for both Bernoulli and Markov modulated loss processes, our estimators are able to infer the underlying packet loss rate at least as accurately as predicted by our theoretical analysis.

In general, we found that, as predicted in our model, the estimation accuracy increases as measurement intervals, sampling rates, and traffic data rates increase. We also found that the accuracy is quite insensitive to the timeout parameters commonly used by NetFlow. Our main conclusions are:

- Both the Fitted and the Weighted methods achieve the expected accuracy as predicted by our analysis in most cases. In particular, we are able to discern a $0.5\%$ packet loss rate over a one minute period under a packet sampling rate of $1/500$ from trace $Backbone$ with a date rate of 2 Gb/sec.
- With the underlying Markov modulated loss process, the empirical results still converge to the average packet loss rates within the predicted accuracy.
- The Weighted method generally provides the best estimation as it utilizes more information.

## V. CONCLUSIONS AND FUTURE WORK

In this paper we proposed correlating flow measurements of the same network traffic taken at different observation points to infer loss on the intervening path. Our technique does not require new router features or measurement infrastructure, and only uses the sampled flow level statistics that are routinely collected in operational networks - which makes this approach easy to deploy and very scalable. The technique is designed to handle challenges of sampled flow-level aggregation such as information aggregation and non-alignment of flow records. We develop multiple loss estimation methods and derive analytical bounds on the variance of estimation. Evaluations using a variety of real traffic traces indicate that the *Fitted* and *Weighted* estimators can accurately estimate the loss rate with low variability and very close to what the analysis predicts.

Our analysis allows us to project the effects on measurement accuracy for future networks. If measurement capacity scales with link rates, the same sampling rate $s$ can be maintained, and from (2) we see that lower link aggregate loss rates can be as accurately measured, since data packets $n$ per unit time increase. But if the measurement capacity does not grow, then $ns$ is constant, so accuracy remains fixed for link aggregates, and is actually worse for any traffic component of a fixed size.

Also, inferring loss rate of a multi-link path is part of the ongoing work.

## REFERENCES

[1] N. Duffield and M. Grossglauser, “Trajectory sampling for direct traffic observation,” *IEEE/ACM Transactions on Networking*, vol. 9, no. 3, pp. 280–292, June 2001.

[2] T. Zseby, “Deployment of sampling methods for sla validation with non-int rusive measurements,” in *PAM*, 2002.

[3] Cisco Systems, “Netflow,” http://www.cisco.com/warp/public/732/netflow/index.html.

[4] M. Schervish, *Theory of Statistics*. New York: Springer, 1995.

[5] M. Yajnik, S. Moon, J. Kurose, and D. Towsley, “Measurement and modeling of the temporal dependence in packet loss,” in *IEEE Infocom’99*.

[6] “NLANR PMA: Special Traces Archive,” http://pma.nlanr.net/Special/.