Technical Report

Department of Computer Science
and Engineering
University of Minnesota
4-192 EECS Building
200 Union Street SE
Minneapolis, MN 55455-0159 USA

TR 01-041

Adaptive Random Sampling for Load Change Detection

Baek-young Choi, Jaesung Park, and Zhi-li Zhang

November 30, 2001

Adaptive Random Sampling for Load Change Detection

Baek-Young Choi, Jaesung Park, Zhi-Li Zhang
Dept. of Computer Science & Engineering
University of Minnesota
Minneapolis, MN55455
{choiby,jpark,zhzhang}@cs.umn.edu

Abstract

Timely detection of changes in traffic is critical for initiating appropriate traffic engineering mechanisms. Accurate measurement of traffic is an essential step towards change detection and traffic engineering. However, *precise* traffic measurement involves inspecting *every* packet traversing a link, resulting in significant overhead, particularly on routers with high speed links. *Sampling* techniques for traffic *estimation* are proposed as a way to limit the measurement overhead. Since the efficacy of change detection depends on the accuracy of traffic estimation, it is necessary to control error in estimation due to sampling. In this paper, we address the problem of *bounding* sampling error within a pre-specified tolerance level. We derive a relationship between the number of samples, the accuracy of estimation and the squared coefficient of variation of packet size distribution. Based on this relationship, we propose an *adaptive random sampling* technique that determines the *minimum* sampling probability adaptively according to traffic dynamics. Using real network traffic traces, we show that the proposed adaptive random sampling technique indeed produces the desired accuracy, while also yielding significant reduction in the amount of traffic samples. We also investigate the impact of sampling errors on the performance of load change detection.

1 Introduction

With the rapid growth of the Internet, traffic engineering has become an important mechanism to reduce network congestion and meet various user demands. Measurement of network traffic load is crucial for configuring, managing, pricing, policing, and engineering the network. The network traffic may fluctuate frequently and often unexpectedly for various reasons such as transitions in user behavior, deployment of new applications, changes in routing policies or failure of network elements. It is a daunting task for network administrators to manually tune the network configuration to accommodate the traffic dynamics. Thus, there is a need for tools that enable intelligent control and management of high speed networks.

Many practical problems arising in network performance monitoring and management are due to the fact that changes in network conditions are observed too late. Moreover, the exact time of a change may not be readily available. Such information on change point can help locate the source of change and initiate an appropriate action to deal with the change. In other words, detection of abrupt changes is an important first step towards reacting to changes by invoking the necessary traffic engineering mechanisms. The problem of change point detection can be addressed using time series analysis of traffic loads. Clearly, *accurate* measurement of traffic is a pre-requisite for identifying the point of change. Most traffic measurement tools require a network device to

capture and store every single packet traversing a link. With today's high-speed links, such an approach is not feasible. It not only taxes the processing capacity of routers or requires special measurement devices, but also generates huge volumes of data that can quickly exhaust storage space. Furthermore, it is extremely time-consuming to process large volumes of captured data, especially if on-line analysis of the data is needed to determine the traffic loads and detect changes in traffic loads on-the-fly. Sampling techniques are therefore a better alternative. However, sampling inevitably introduces errors in the traffic load estimation. Such errors may adversely affect the change point detection of traffic loads.

In this paper we develop an *adaptive random sampling* technique for load change detection using *sampled traffic measurement*. Our adaptive random sampling technique differs from existing sampling techniques for traffic measurement in that it yields *bounded* sampling errors *within a pre-specified error tolerance level*. Such error bounds are important in reducing the "noise" in change point detection with sampled traffic measurement. Furthermore, the pre-specified error tolerance level allows us to control the performance of load change detection algorithms as well as the amount of packets sampled. The paper is devoted to the analysis and verification of the proposed adaptive random sampling technique and the impact of sampling errors on the performance of traffic load change detection. Our contributions are summarized as follows.

We observe that sampling errors in estimating traffic load arises from dynamics of packet sizes and counts, and these traffic parameters vary over time. Consequently, *static* sampling (i.e., with a fixed sampling rate) cannot guarantee errors within a given error tolerance level. From analysis, we find that the *minimum* required number of samples to bound sampling error within a given tolerance level is proportional to the squared coefficient of variation (SCV) of packet size distribution. Using this relationship, we propose an adaptive random sampling technique that determines the (minimum) sampling probability adaptively based on the SCV of packet size distribution and the packet count. More specifically, time is divided into (non-overlapping) observation periods (referred to as (time) blocks), and packets are sampled in each observation period. At the end of each block, in addition to estimating the traffic volume of the block, the SCV of packet size distribution and the packet count of the block are calculated using the traffic samples. These traffic parameters are used to predict the SCV of packet size distribution and the packet count of the next block, using an *Auto-regressive* (AR) model. The sampling probability for the next block is then determined based on these predicted values and the given error tolerance level. The procedure is depicted in Figure 1. Through analysis, we quantify the estimation and prediction errors introduced by our sampling technique, and devise mechanisms to control their impact on the traffic load estimation. Using real network traffic traces, we show that the proposed adaptive random sampling technique indeed produces the desired accuracy, while at the same time yielding significant reduction in the amount of traffic samples. For the time series of estimated traffic load, we present a non-parametric on-line change point detection algorithm based on singular value spectrum analysis. The algorithm finds nonstationarities in traffic loads at some larger, configurable operational time scale using sampled measurements obtained at each (smaller time-scale) observation period. The basic approach is depicted in Figure 2. We investigate the impact of sampling errors on the performance of this load change detection algorithm using real network traffic traces.

Before we leave this section, we would like to comment that in the context of traffic measurement and analysis, several sampling methods have been proposed and studied for various applications. Statistical sampling of network traffic was first used in [10] for measuring traffic on the NSFNET backbone in the early 1990's. Claffy *et al.* evaluated classical event and time driven *static* sampling methods to estimate statistics of distributions of packet size and inter-arrival time. Trajectory sampling proposed in [5] directly observes the entire traffic traversing through a network domain,

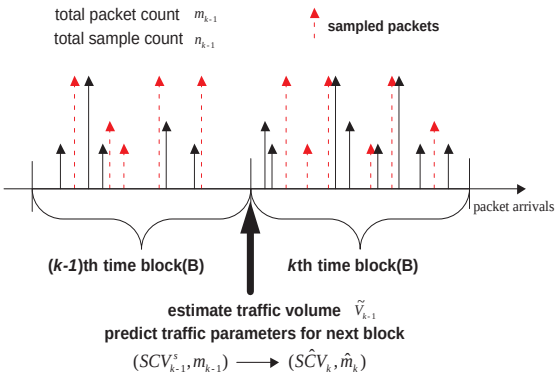


Figure 1: Adaptive random sampling.

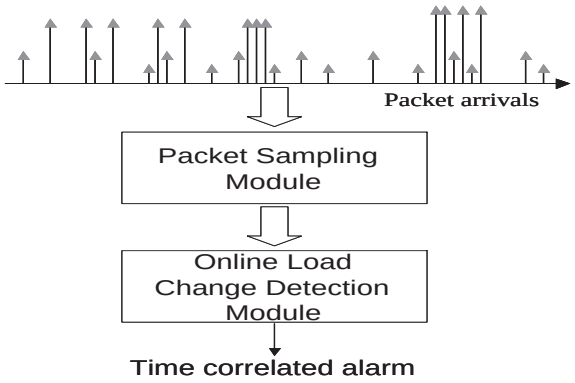


Figure 2: System model.

and infers statistics on the *spatial* relations of the network traffic. A size-dependent *flow* sampling method is proposed in [1] for the purpose of usage-sensitive charging. In [2], the problem of identifying large flows is studied. A probabilistic packet sampling method is used to identify large flows and sampling probability is computed for each packet based on its size. This method requires each packet header to be inspected. None of these sampling techniques address the issue of bounding sampling errors in random packet sampling, and thus cannot be applied to change point detection with sampled traffic loads.

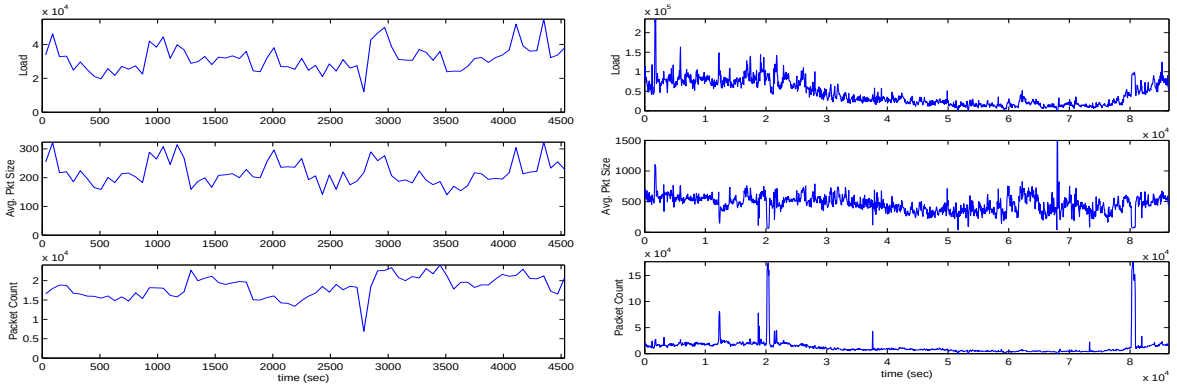
The remainder of the paper is structured as follows. In Section 2, we formally state the problem addressed in this paper. In Section 3, the adaptive random sampling technique is described and analyzed. Experimental results with real network traffic traces are presented in Section 4. We present the change point detection algorithm with sampled measurement in Section 5. Section 6 concludes the paper.

2 Sampling Problem for Load Change Detection

In this section we first formulate the sampling problem for detecting abrupt changes in traffic loads. We then derive a lower bound on the number of samples needed to estimate the traffic load accurately within a given tolerance level. Based on this, we determine the sampling probability that is *optimal* in the sense that it guarantees the given accuracy with the minimum number of samples. The optimal sampling probability depends on both the number of packets and the variation in their sizes in an observation period. We see that the network traffic fluctuates significantly over time in terms of both the number of packets and their sizes. Hence, the optimal sampling probability also varies over time. This suggests that *static* sampling with *fixed* sampling probability may result in either erroneous undersampling or unnecessary oversampling. In other words, static sampling cannot capture the traffic dynamics accurately or efficiently. This motivates us to develop an *adaptive* random sampling technique that attempts to minimize the sampling frequency while ensuring that the sampling error is bounded.

2.1 Bounding Sampling Errors in Traffic Load Estimation

Traffic load is the sum of the sizes of packets arriving during a certain time interval. Thus, traffic load is determined by both the number of packets and their sizes. In determining the traffic load,



(a) Auckland-II trace (Π_4)

(b) Auckland-II trace (Π_5)

Figure 3: Impact of packet size and packet count on traffic load.

the variability of packet sizes is often overlooked and only packet count is considered. However, as noted in [20], average packet size plays an important role in estimating the traffic load. Consider, for example, two network traffic traces captured at University of Auckland [8] to US link. The time series plots of the traffic loads of the two traces (Π_4 and Π_5 in Table 2 are shown in the top row of both Figure 3(a) and 3(b). The plots in the middle row show the average packet sizes over time, while the plots in the bottom row show the packet counts over time. From Figure 3(a), we see that the increase in the traffic load around 1000 sec is due to the increase in the packet size rather than the packet count. On the other hand, the abrupt increase in the packet count near 2×10^4 sec in Figure 3(b) does not lead to any increase in the traffic load, since the packet sizes at the time are extremely small. These examples illustrate that the variation in packet sizes is an important factor in estimating the traffic load using sampling. In fact, we will show later that the variation in packet sizes is the key factor in determining the sampling rate and for controlling the accuracy of load estimation.

The reason that we highlight the factors that affect the traffic load estimation using sampling is that *accurate* estimation of traffic load is crucial in detecting changes in traffic loads. For change point detection, the series of the estimated traffic loads must retain the *change* or *stability* of the original traffic. Significant sampling errors in traffic load estimation can distort the original "signal" and lead to *false alerts* that may adversely affect the performance of networks, for instance, if they inadvertently trigger inappropriate traffic engineering mechanisms. Hence quantifying and bounding sampling errors is critical in applying sampling techniques to traffic load estimation for the purpose of load change detection.

Time series analysis requires that observations be uniformly spaced in time. Packet arrivals at a link in the Internet are by nature irregularly spaced in time and so are the packet samples. To obtain a uniformly spaced time series, traffic loads can be estimated from packets sampled during (non-overlapping) observation periods of fixed length (see Figure 1). We refer to an observation period as a (load estimation) *time block*, or simply *block*. The length of a block is denoted by B, which can be configured depending on the specific engineering purposes. To preserve the trend of the original traffic load, the sampling error in each block must be bounded quantitatively. In the following we state the problem of bounding sampling errors in traffic load estimation formally.

Assume that there are m packets arriving in a block, and let X_i be the size of the ith packet. Hence the traffic load of this block is $V = \sum_{i=i}^{m} X_i$. To estimate the traffic load of the block, suppose we *randomly* sample n, $1 \leq n \leq m$, packets out of the m packets. In other words, each packet has an equal probability $p = n/m$ to be sampled. Let $\hat{X}_j$, $j = 1, 2, \ldots, n$, denote the size of the jth sampled packet. Then the traffic load V can be estimated by $\hat{V}$ using the samples, where where $\hat{V}$ is given by

$$\hat{V} = \frac{m}{n} \sum_{j=1}^{n} \hat{X}_j \tag{1}$$

It can be shown that $\hat{V}$ is an unbiased estimator of V, i.e., $E[\hat{V}] - V = 0$.

Our objective is to bound the relative error $\left|\frac{\hat{V}-V}{V}\right|$ within a *prescribed* error tolerance level given by two parameters $\{\eta, \varepsilon\}$ $(0 < \eta, \varepsilon < 1)$, i.e.,

$$Pr\left\{\left|\frac{\hat{V}-V}{V}\right| > \varepsilon\right\} \leq \eta. \tag{2}$$

In other words, we want the relative error in traffic load estimation using random sampling to be bounded by ε with a high probability $1 - \eta$. Given this formulation of the bounded error sampling problem, the question is *what is the minimum number of packets that must be sampled randomly so as to guarantee the prescribed accuracy.* We address this question in the following subsection.

2.2 Optimal Sampling Probability and Limitations of Static Sampling

From the *central limit theorem of random samples* [3], as the sample size $n \to \infty$, the average of sampled data approaches the population mean, regardless of distribution of population. Thus (2) can be rewritten as follows:

$$Pr\left\{\left|\frac{\hat{V}-V}{V}\right| > \varepsilon\right\} = Pr\left\{\frac{\sqrt{n}}{\sigma}\left|\frac{1}{n}\sum_{i=1}^{n} X_i - \mu\right| > \frac{\varepsilon\mu\sqrt{n}}{\sigma}\right\} \approx 2\left(1 - \Phi\left(\frac{\varepsilon\mu\sqrt{n}}{\sigma}\right)\right) \leq \eta, \tag{3}$$

where μ and σ are, respectively, the population mean and standard deviation of the packet size distribution in a block, and $\Phi(\cdot)$ is the cumulative distribution function (c.d.f) of the standard normal distribution (i.e.,$N(0, 1)$). Hence, to satisfy the given error tolerance level, the required number of packet samples must satisfy

$$n \geq n^* = \left(\frac{\Phi^{-1}(1-\eta/2)}{\varepsilon} \cdot \frac{\sigma}{\mu}\right)^2 = z_p \cdot S \tag{4}$$

where $z_p = \left(\frac{\Phi^{-1}(1-\eta/2)}{\varepsilon}\right)^2$ and $S = (\sigma/\mu)^2$ is the squared coefficient of variance (SCV) of the packet size distribution in a block. Eq. (4) concisely relates the minimum number of packet samples to the estimation accuracy and the variability in packet sizes. In particular, it states the minimum required number of packet samples, n^*, is *linearly* proportional to the squared coefficient of variance, S, of the packet size distribution in a block.

From (4) we conclude that the *optimal* sampling probability, p^*, which samples the minimum required number of packets in a block, is given by

$$p^* = \frac{n^*}{m}. \tag{5}$$

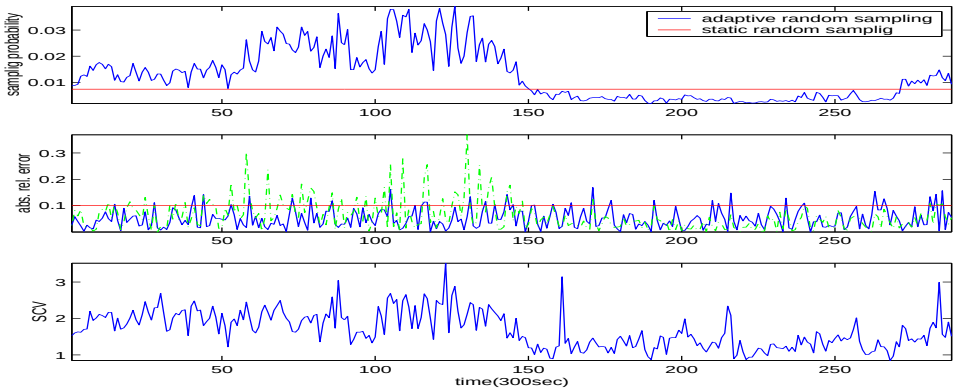


Figure 4: Adaptive random sampling vs. static random sampling ($\{\eta, \varepsilon\} = \{0.1, 0.1\}$).

Hence, to attain the prescribed sampling accuracy $\{\eta, \varepsilon\}$, packets in a block must be sampled randomly with a probability at least p^*. Note that to determine the optimal sampling probability p^*, we need to know the *actual* SCV of the packet size distribution and the packet count m in a block. Unfortunately, in practice these traffic parameters of a block are *unknown* to us at the time the sampling probability for the block must be determined. To circumvent this problem, in Section 3 we develop an AR (Auto-regressive) model to predict these parameters of a block based on past sampled measurements of previous blocks. Before we proceed to present this model, we would like to conclude this section by discussing the limitations of *static* sampling.

Static sampling techniques such as "one-out-of-N" sampling are commonly employed in routers, as they are simple to implement. For example, Cisco's Sampled NetFlow [7] introduced in IOS 12.0(3)T samples one packet out of every N IP packets for flow statistics. More generally, static *random* sampling technique randomly samples a packet with a *fixed* probability. Both techniques do not take traffic load dynamics into account, thus when applied to traffic load estimation, they cannot guarantee that the sampling error in each block falls within a prescribed error tolerance level. Furthermore, it is difficult to determine what is the appropriate fixed sampling probability (or the value for N in "one-out-of-N" sampling) to be used for all blocks.

To help illustrate the importance of adjusting sampling probability to packet size variability, in Figure 4 we compare the *optimal* adaptive random sampling technique to the static random sampling technique using the Auckland trace Π_1 shown in Table 2. To make fair comparison, the fixed sampling probability for the static random sampling technique is set such that the *sampling fraction* (i.e., the amount of sampled data) over the entire trace is the same as that under the optimal adaptive random sampling technique. The top plot in Figure 4 shows the optimal sampling probability used by the adaptive sampling technique over time (the block size $B = 300sec$) as well as the fixed sampling probability used by the static random sampling. The middle plot shows the resulting relative errors by both sampling techniques. The bottom plot shows the SCV of the packet sizes across the blocks.

From the figure we see that when the variability of packet size distribution of a block is large, static random sampling tends to *undersample* packets, resulting in large estimation errors. This may lead to false alarm or non-detection by a load change detection algorithm. On the other hand, when the variability of packet size distribution of a block is small, static random sampling tends to *oversample* packets, thereby wasting processing capacity and memory space of the measurement device. Moreover, the frequent oscillation between oversampling and undersampling of static random

Table 1: Notation.

S_k	SCV of the population of kth block
S_k^s	SCV of the samples of kth block
$\hat{S}_k^s$	predicted SCV of the samples of kth block
n_k^*	minimum number of samples needed in kth block
$\hat{n}_k$	predicted minimum number of samples needed in kth block
$\tilde{n}_k$	actual number of samples in kth block
m_k	actual number of packets in kth block
$\hat{m}_k$	predicted number of packets in kth block

sampling causes undesirable increase in the variance of estimation errors. This example demonstrates that in order to ensure a desired accuracy in traffic load estimation while without resorting to unnecessary oversampling, packet sampling probability for each block must be adjusted in accordance with the traffic load dynamics. This is the essential idea behind our proposed adaptive random sampling technique. The key challenge remains to be addressed is how to determine the (optimal) sampling probability for each block *without a priori knowledge* of the traffic parameters – the SCV of packet size distribution and packet count of a block. The next section is devoted to the analysis and solution of this problem.

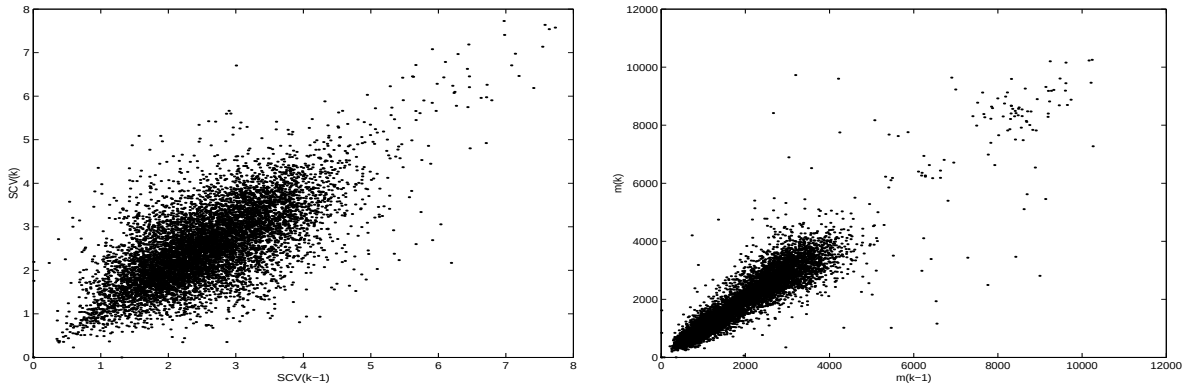
3 Adaptive Random Sampling with Bounded Errors

In this section we present an AR (Auto-regressive) model for predicting two key traffic parameters for traffic load estimation – the SCV of packet size distribution and packet count of a block – using past (sampled) data from previous blocks. The AR model is justified by empirical studies using real network traffic traces. In addition to estimation errors due to sampling, the prediction model also introduces *prediction* errors. We quantify and analyze the impact of these errors on the traffic load estimation and discuss how these errors can be controlled.

3.1 AR Model for Traffic Parameter Prediction

The efficacy of prediction depends on the correlation among the past and future values of the parameters being predicted. We have analyzed many public-domain real network traffic traces, a subset of traces we studied is listed in Table 2. We found that the SCV's of the packet sizes of two *consecutive* blocks are strongly correlated; the same is also true for the packet counts, m's, of two *consecutive* blocks. As an illustration, Figures 5(a) and 5(b) show, respectively, the scatter plots of SCV and m of two consecutive blocks (the block size $B = 60sec$) using the trace Π_4 in Table 2. It is evident that the values of SCV and m of two consecutive blocks are highly correlated. In fact, there is a strong *linear* relationship between these values.

As a further justification, we remark that the predictability of network traffic has also been studied by other researchers. For instance, in [11] the authors investigated the questions of how far into the future a traffic rate process can be predicted for a given error constraint, and how much the prediction error is over a specified number of future time intervals (or steps). They showed that prediction works well for one step into the future, although the prediction accuracy degrades



(a) SCV

(b) Packet count

Figure 5: Relationship between past and future values of SCV and packet count.

quickly as the number of steps increases. In the context of our work, note that we only need to predict the traffic parameters for the next step (i.e., the next block).

The strong linear relationship evident in Figures 5(a) and 5(b) suggests that linear regression can be used for the prediction of the SCV of packet sizes and packet count m of a future block using the values of the previous blocks. We employ an AR (Auto-regressive) model for predicting the traffic parameters SCV and m, as compared to other time series models, the AR model is easier to understand and computationally more efficient. In particular, using the AR model, the model parameters can be obtained by solving a set of simple linear equations [4], making it suitable for online traffic load estimation. In the following we formally describe the AR model for the traffic parameter prediction.

We first present an AR(u) model for predicting the SCV of the next block using the SCV of *sampled* packet sizes of the u previous blocks. The notation used here and in the rest of this paper is summarized in Table 1. Let S_k be the SCV of the packet sizes in the kth block, and S_k^s be the SCV of the *packet sizes randomly sampled* in the kth block. We can relate S_k and S_k^s as follows:

$$S_k^s = S_k + Z_k \tag{6}$$

where Z_k denotes the error in estimating the actual SCV of the packet sizes using the random packet samples. (We refer to Z_k as the estimation error.)

Using the AR(u) model [4], S_k^s can be expressed as

$$S_k^s = \sum_{i=1}^{u} a_i^s S_{k-i}^s + e_k^s \tag{7}$$

where a_i, $i = 1, \ldots, u$, are the model parameters, and e_k^s is the *uncorrelated* error (which we refer to as the *prediction error*). The error term e_k^s follows a normal distribution with mean 0 and variance $var(e_k^s) = \sigma_{S_k^s}^2 (1 - \sum_{i=1}^{u} a_i^s \rho_{S_k^s,i})$. Here $\rho_{S_k^s,i}$ is the lag-i autocorrelation of S_k^s's. The model parameters a_i, $i = 1, \ldots, u$, can be determined by solving a set of linear equations (8) in terms of v past values of S_i^s's, where $v \geq 1$ is a configurable parameter independent of u, and is typically

referred to as the memory size.

$$\rho_h = \sum_{i=1}^{u} a_i \rho_{h-i}, \text{ where } h = v, \ldots, v-u+1 \text{ and } \rho_h \text{ is lag-}h \text{ autocorrelation of the data} \quad (8)$$

Using the above AR(u) model, at the end of the $(k-1)$th block, we predict the SCV of the kth block using the SCV values of the sampled packet sizes of the u previous blocks as follows:

$$\hat{S}_k^s = \sum_{i=1}^{u} a_i^s S_{k-i}^s. \quad (9)$$

Combining (6), (7) and (9), we have

$$\hat{S}_k^s = S_k + Z_k + e_k^s. \quad (10)$$

Hence we see that there are two types of errors in predicting the actual SCV of the packet size of the next block using the sampled packet sizes of the previous blocks: the estimation error Z_k due to random sampling, and the prediction e_k^s introduced by the prediction model. The total resulting error is $Z_k + e_k^s$. In Section 3.2 we analyze the properties of these errors and their impact on the traffic load estimation.

We now briefly describe how the packet count m_k of the kth block can be estimated based on the past packet counts using the AR(u) model. Let m_k denote the packet count of the kth block, then using the AR(u) model, we have $m_k = \sum_{i=1}^{u} b_i m_{k-i} + e_{m,k}$, where as before b_i, $i = 1, 2, \ldots, u$, are the model parameters, and $e_{m,k}$ is the prediction error term, which is normally distributed with zero mean. Let $\hat{m}_k$ denote the *predicted* packet count of the kth block. Using the the AR(u) prediction model, we have $\hat{m}_k = \sum_{i=1}^{u} b_i \hat{m}_{k-i}$.

As in the case of predicting SCV of the packet sizes using the AR(u) prediction model, the prediction of the packet count using the past *sampled* packet counts introduces both estimation error and prediction error. However, in the case of predicting the packet count m, it is *not* unreasonable to assume that the *actual* packet count of a block is known at the end of the block. This is because in the modern commercial router design, a packet counter is often included in the line card of a router, as such a packet counter does not overly burden a router in terms of both processing and memory capacities[1].

In this case, we can predict the packet count of the next block using the *actual* packet counts of the previous blocks. Namely, $\hat{m}_k = \sum_{i=1}^{u} b_i m_{k-i}$. Hence only the prediction error is involved when a packet counter is available. For simplicity, we will assume that this is the case our paper. (Note that this assumption does not change the nature of the adaptive random sampling technique we proposed, only simplifying the analysis of the sampling errors.) Given the predicted SVC of the packet size distribution and packet count of the next block, we can now calculate the (predicted) minimum number of required packet samples using (4) and the sampling probability for the next block:

$$\hat{n}_k = z_p \hat{S}_k^s \text{ and } \hat{p}_k = \frac{\hat{n}_k}{\hat{m}_k}. \quad (11)$$

[1]Observe the packet count of a block can be collected without inspecting the contents of a packet. Hence it does not cause significant burden on routers. For example, consider a link with bandwidth $10Gbps$. Suppose the worst case where only the smallest IP packets (40 bytes) are arrived. Then, there can be at most $1.875G$ packets in a block of 60 seconds. The size of counter needed is only 32 bits. If we assume that I instructions are needed to increment the counter, then we need only $31.25 * I$ MIPS for maintaining the packet counter.

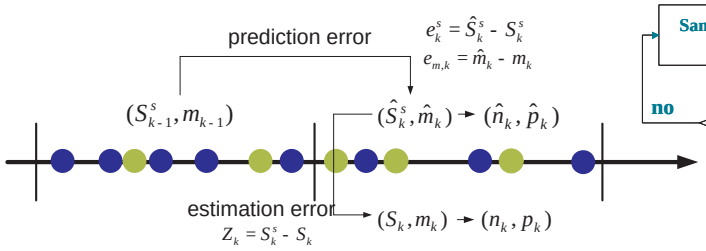


Figure 6: Traffic parameter prediction process.

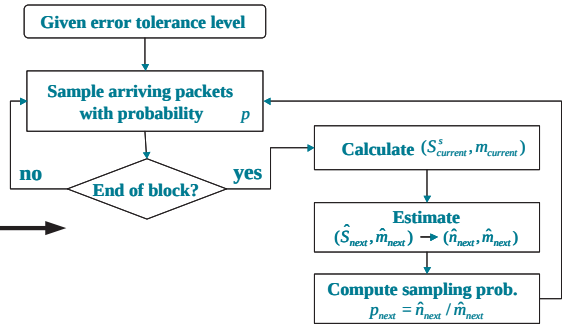
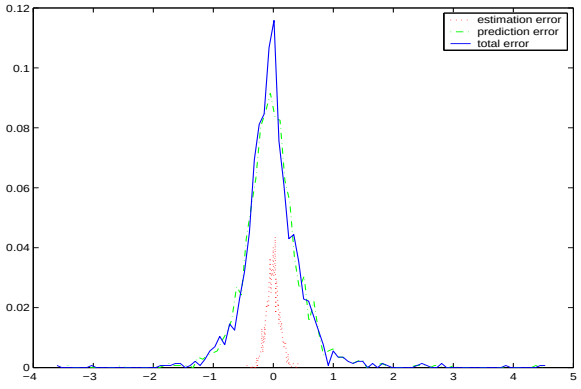
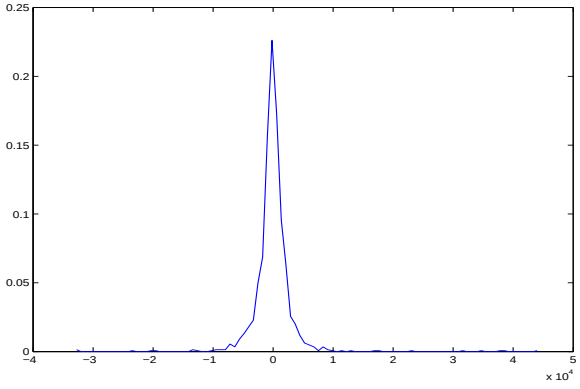


Figure 7: Flow chart of adaptive random sampling.



(a) Error in SCV prediction and estimation.



(b) Error in m prediction.

Figure 8: Gaussian prediction error.

The entire process of predicting traffic parameters SCV and m is depicted in Figure 6. Figure 7 shows the flow chart of the adaptive random sampling procedure. Using the AR prediction model, at the end of each block, the model parameters (a_i's for SCV, b_i's for m) need to be computed. The complexity of the AR prediction model parameter computation is only $O(v)$ where v is the memory size. Through empirical studies, we have found that small values of the memory size (around 5) are sufficient to yield good prediction. Figure 9 depicts average AR predition error compared to memory size using the trace Π_1 (the block size $B = 300sec$).

3.2 Analysis of Errors in Traffic Load Estimation via Sampling

In this subsection we analyze the impact of estimation and prediction errors on the traffic load estimation. We first study the properties of the errors introduced by the adaptive random sampling process. We then establish several lemmas and theorems to quantify the impact of these errors on the relative error in the traffic load estimation.

Recall from (10) that there are two types of errors in estimating the SCV of the packets size of the next block using the past sampled packet sizes: the estimation error Z_k and the prediction error e_k^s. From empirical studies using real network traces, we have found that the errors generally

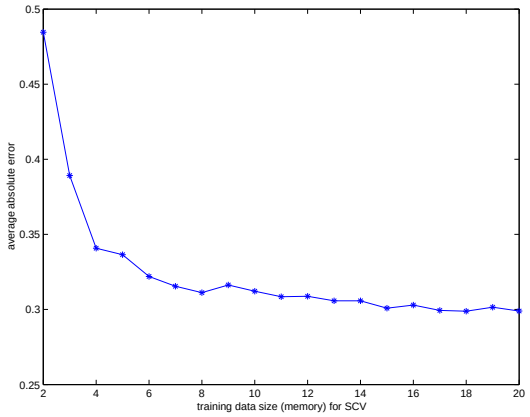


Figure 9: AR prediction error vs. training memory size.

follow a normal distribution with mean 0. An example using the trace Π_1 is shown in Figure 8(a), we see that both the estimation error and prediction error as well as the total error $(Z_k + e_k^s)$ have a Bell-shape centered at 0. We have performed the skewness test and kurtosis test [19], and these tests conform the normality of these errors. Similar empirical studies have also shown that the error $(e_{m,k})$ in the packet count prediction is also normally distributed with zero mean. See Figure 8(b) for an example using the same network traffic trace as in Figure 8(a).

The above results suggest that we can approximate both the estimation error and prediction error using normal distributions with zero mean. This allows us to quantify the variance of the errors introduced by the adaptive random sampling process. For example, assume, for simplicity , that an AR(1) model is used for predicting S_k, the SCV of the packet sizes of the kth block. Then the variance of the prediction error, $var(e_k^s)$, is given by $var(e_k^s) = \sigma^2_{S_k^s}(1 - a_1^s \rho_{S_k^s,1})$, where $\rho_{S_k^s,1}$ is the lag-1 autocorrelation of S_k^s. From (6) and (7), we have $var(Z_k) = (a_1^s)^2 var(S_{k-1}^s) + var(e_k^s) - var(S_k)$. Given sufficient packet samples, $var(S_{k-1}^s) \approx var(S_k)$. Thus $var(Z_k) = \sigma^2_{S_k^s}((a_1^s)^2 - a_1^s \rho_{S_k^s,1})$. Therefore the variance of the total error in predicting S_k is

$$var(Z_k) + var(e_k^s) = \sigma^2_{S_k^s}(1 - 2a_1^s \rho_{S_k^s,1} + (a_1^s)^2). \tag{12}$$

We now quantify the impact of these errors on the relative error in the traffic load estimation. Define $\tilde{n}_k = m_k \cdot \frac{\hat{n}_k}{\hat{m}_k}$, which is the *actual* number of packets randomly sampled (on the average) in the kth block, given the (predicted) minimum sampling probability $\hat{p}_k = \hat{n}_k / \hat{m}_k$. Then the estimated traffic load of the kth block is given

$$\tilde{V}_k = \frac{m_k}{\tilde{n}_k} \sum_{j=1}^{\tilde{n}_k} \hat{X}_j, \tag{13}$$

where $\hat{X}_j$ denotes the packet size of the jth randomly sampled packet in the kth block.

Using the central limit theorem for a sum of a random number of random variables (see p.369, problem 27.14 in [6]), we can establish the following two lemma and theorem. The proofs can be found in the appendix.

Lemma 1 $\frac{\tilde{n}_k}{n_k^*}$ *converges to 1 almost surely as* $n_k^* \to \infty$.

Theorem 2 *With probability $1-\eta$, the relative error in estimating the traffic load V_k of the kth block is*

$$\begin{aligned}\left|\frac{\tilde{V}_k - V_k}{V_k}\right| &\leq \varepsilon + \frac{1}{\sqrt{z_p}}(1+\varepsilon)Y + o(\frac{1}{m}) \\ &\approx \varepsilon + \frac{1}{\sqrt{z_p}}(1+\varepsilon)Y\end{aligned}$$

where recall that $z_p = \left(\frac{\Phi^{-1}(1-\eta/2)}{\varepsilon}\right)^2$, and Y is a normally distributed random variable with mean 0 and variance 1, i.e., $Y \sim N(0,1)$.

Theorem 2 yields a theoretic bound on the variance of adaptive random sampling, i.e.,

$$var\left(\left|\frac{\tilde{V}-V}{V}\right|\right) \leq \frac{(1+\varepsilon)^2}{z_p} \text{ with probability } 1-\eta. \tag{14}$$

Notice that the variance of adaptive random sampling is independent of the distribution of objects being sampled and is *controllable* by the accuracy parameter. On the other hand, the variance of static random sampling depends on the SCV and the number of samples. i.e.,

$$\begin{aligned}var\left(\frac{\hat{V}-V}{V}\right) &= var\left(\frac{\frac{m}{n}\sum_{i=1}^{n}\hat{X}_i - \sum_{j=1}^{m}X_j}{\sum_{j=1}^{m}X_j}\right) = var\left(\frac{\sum_{i=1}^{n}\hat{X}_i}{n\mu}\right) \\ &= \left(\frac{1}{n\mu}\right)^2 \cdot n \cdot \sigma^2 = \frac{\sigma^2}{\mu^2}\cdot\frac{1}{n} = \frac{S}{n}\end{aligned} \tag{15}$$

The variance bound (14) of adaptive random sampling suggests that in order to accommodate the prediction and estimation errors introduced by the traffic parameter predictions, we can replace the error bound ε by a tighter bound ε':

$$\varepsilon' = \varepsilon - s \cdot \frac{(1+\varepsilon)}{\sqrt{z_p}} \tag{16}$$

where s is a small adjustment parameter that can be used to control the variance of the relative error.

4 Empirical Evaluation

In this section we empirically evaluate the performance of our adaptive random sampling technique using the real network traces. The traces used in this study are obtained from NLANR [8], and their statistics are listed in Table 2. In this study we have primarily used the *long* duration traces (the Auckland-II traces) to produce more sound statistics and reliable results. But we have also investigated the short duration traces from the higher speed links. We believe that the efficacy of our adaptive random sampling technique as demonstrated in this section are applicable to other traces. For consistency of illustration, the results shown in this section are based on the trace Π_1 unless otherwise specified.

Table 2: Summary of traces used.

Trace name	Trace	Arrival rate	Duration
Π_1	Auckland-II 19991201-192548-0	92.49KBps	24h 02m 58sec
Π_2	Auckland-II 19991201-192548-1	55.16KBps	24h 02m 57sec
Π_3	Auckland-II 19991209-151701-1	49KBps	23h 11m 38sec
Π_4	Auckland-II 20000117-095016-0	168KBps	2h 23m 15sec
Π_5	Auckland-II 20000114-125102-0	222.14KBps	21m 37sec
Π_6	AIX (OC12c) 989950026-1	25.36MBps	90sec
Π_7	AIX (OC12c) 20010801-996689287-1	21.60MBps	90sec
Π_8	COS (OC3c) 983398787-1	4.95MBps	90sec

To show the effectiveness of the prediction model used in our adaptive random sampling technique, we first compare the performance our technique with that of the *ideal* optimal sampling. In the ideal optimal sampling, the optimal sampling probability for each block is computed using (5), assuming that the SCV of the packet sizes and packet count of the block is known. The results are shown in Figure 10. The figure on the top shows the time series of the original traffic load, the estimated traffic loads using both the ideal optimal sampling and the adaptive random sampling with prediction. For the accuracy parameters of $\{\eta, \varepsilon\} = \{0.1, 0.1\}$, the series are very close and hardly differentiable visually. The figure on the bottom shows the cumulative probability of relative errors in traffic load estimation for both the ideal optimal sampling and adaptive random sampling with prediction. The horizontal line in the figure indicates the $(1-\eta)$th quantile of the errors. We see that for both the sampling methods, the traffic load estimation indeed conforms to the pre-specified accuracy parameter, i.e., the probability of relative errors larger than $\varepsilon = 0.1$ is around $\eta = 0.1$.

To further investigate the performance of the adaptive random sampling with prediction, in Figure 11 we vary the error bound ε (while fixing η at 0.1), and plot the corresponding $(1-\eta)$th quantile of relative errors. We see that the $(1-\eta)$th quantiles of relative errors for the whole range of the error bound η stay close to the prescribed error bound. For comparison, in the figure we also plot the corresponding results obtained using the static random sampling. Here to provide fair comparison, the (fixed) sampling probability of the static random sampling is chosen such as the *sampling fraction* (or, the total amount of sampled data) over the entire trace is the same as that of the adaptive random sampling. We see that for all range of the error bound, the static random sampling produces a much larger the $(1-\eta)$th quantile of relative errors.

Another key metric for comparing sampling techniques is the variance of an estimator [18]. Small variance in estimation is a desired feature of a sampling method in that the estimate is more reliable when used in place of the value of a population. This feature is especially important when the sampling method is applied to change point detection, since large variation in estimation may cause outliers in the estimated signal, making it difficult to detect change points (see discussion in Section 5). In Figure 12 we compare the standard deviation of the relative errors in traffic load estimation for both the adaptive random sampling and the static random sampling. As the figure shows, the variation of errors of the adaptive random sampling is always bounded within the theoretic upper bound (14). On the contrary, due to frequent excessive undersampling and oversampling (as noted in Section 2), the static random sampling has a much larger variation of errors. In particular, the error variance of the static random sampling is always larger than

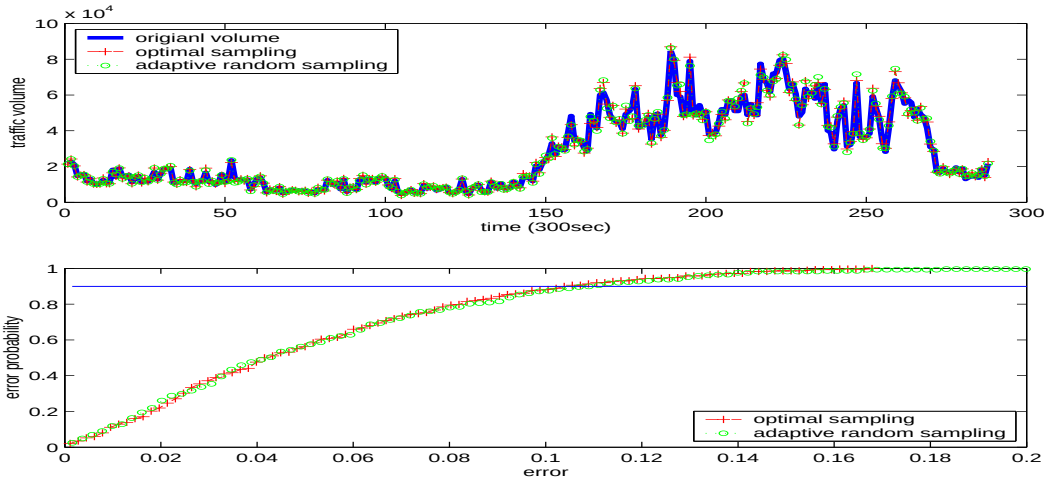


Figure 10: Traffic volume estimations and relative error ($\{\eta, \varepsilon\} = \{0.1, 0.1\}, B = 300sec$).

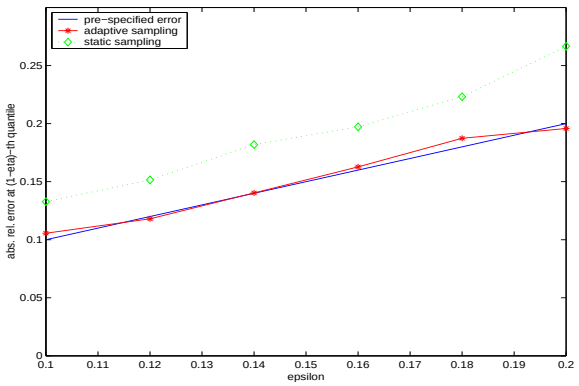


Figure 11: $(1-\eta)$th quantile relative error ($\eta = 0.1, B = 60sec$).

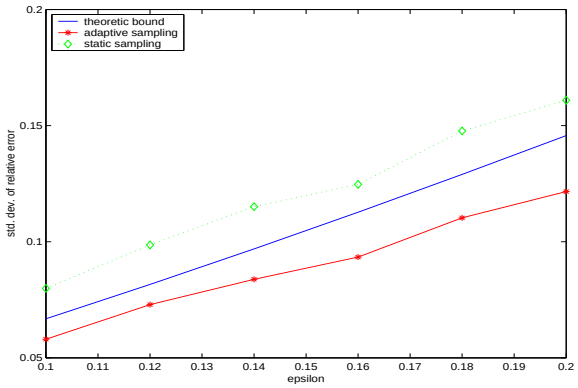
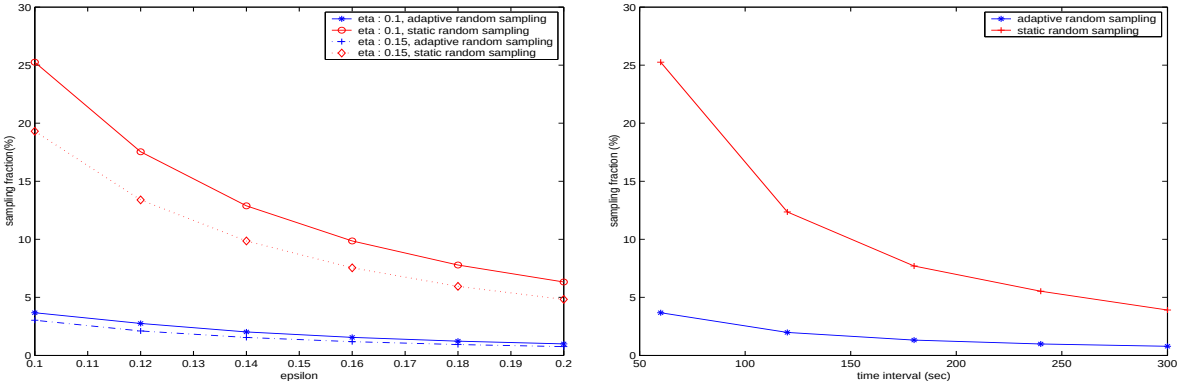


Figure 12: Standard deviation of relative error ($\eta = 0.1, B = 60sec$).

the theoretic variance bound for the adaptive random sampling. In summary, the above results demonstrate the superior performance of our adaptive random sampling technique over the static random sampling.

We now compare the adaptive random sampling and static random sampling in terms of their *resource efficiency*. We measure the resource efficiency using the sampling fraction – the ratio of the total amount of sampled data produced by a sampling technique over the total amount data in a trace. Sampling fraction provides an indirect measure of the processing and storage requirement of a sampling technique. To compare the adaptive random sampling and static random sampling, we choose the (fixed) sampling probability for the static random sampling in such a manner that the $(1-\eta)$th quantile of relative errors satisfies the same error bound as the adaptive random sampling. Figure 13 shows the sampling fraction of the two sampling methods as we vary the error bound ε. For both methods, tighter error bound requires more packets to be sampled. However, for the same error bound, the adaptive random sampling requires far fewer packets to be sampled overall. Figure 13(b) shows the impact of time block size B on the sampling fraction. For both sampling methods, as the time block size increases, the sampling fraction decreases. This is because the estimation accuracy is determined by the number of required packet samples, which is independent



(a) Different accuracy parameters ($B = 60sec$)

(b) Different time interval

Figure 13: Sampling fraction ($\{\eta, \varepsilon\} = \{0.1, 0.1\}$).

of the number of packet arrivals. As the time block size increases, fewer packet samples are needed *relative to the total number of packet arrivals* to achieve the estimation accuracy, resulting in a smaller sampling fraction. Although a larger time block yields faster decrease in the sampling fraction for the static random sampling, even with a block size of 300 seconds (5 minutes), the sampling fraction of the adaptive random sampling is still several times smaller than the static random sampling. Note that the average data rate of the trace Π_1 (used in the studies shown in the figures) is less than 1 MBps. It is not hard to see that in highly loaded links and high speed links where the traffic load fluctuates more frequently, the adaptive random sampling will lead to more gains in terms of the sampled data reduction (i.e., smaller sampling fraction). To illustrate this, we apply our adaptive random sampling technique to the trace Π_6 which has an average data rate of 35.36 Mpbs. For the accuracy parameters $\{0.1, 0.1\}$ and a block size of 30 seconds, the resulting sampling fraction is only 0.022%!

To conclude this section, we provide a more detailed study of the SCV of packet sizes in the real network traffic traces. Understanding of the SCV of packet size distribution is important, as the number of required packet sizes is proportional to SCV. Thus the SCV of packet sizes in a network traffic trace has a direct impact on the resulting sampling fraction. The SCV statistics of the traces are presented in Table 3. We see that the SCVs of packet sizes of the traces vary significantly, although some of the traces (Π_1 - Π_5) are captured over the same physical link over different time. For the accuracy parameters of $\{\eta, \varepsilon\} = \{0.1, 0.1\}$, the sampling fractions for these traces (with block size $B = 60sec$ or $300sec$) are also listed in Table 3. It is clear that the SCV of packet sizes of the traces has a direct impact on the sampling fraction. In general, smaller SCV leads to smaller sampling fraction. Furthermore, the data arrival rate of the traces also affects the sampling fraction. For example, the traces Π_6 and Π_8 have similar SCV's. However, the average data rate of Π_6 is about five times faster than that of Π_8 (see Table 2). As a result, the sampling fraction of Π_6 is about 5 times smaller than that of Π_8.

Finally, Figure 14 shows the relation between SCV and traffic load in a scatter plot using the traces. We observe that when the link is more highly utilized (i.e., larger traffic load), the SCV of packet sizes tends to be smaller. This seems to indicate that more packets of similar sizes are arriving on the link. Since the lower SCV leads to a smaller number of required packet samples,

Table 3: SCV variability and sampling fraction ($\{\eta, \varepsilon\} = \{0.1, 0.1\}$).

Trace	avg.		min.		max.		sampling fraction (%)	
	$B = 60s$	$B = 300s$	$B = 60s$	$B = 300s$	$B = 60s$	$B = 300s$	$B = 60s$	$B = 300s$
Π_1	1.70	1.68	0.69	0.85	5.21	3.52	5.48	0.67
Π_2	2.11	2.09	0.39	0.60	5.29	4.32	5.85	0.83
Π_3	2.59	2.48	0.47	0.90	7.39	6.44	5.07	1.01
Π_4	1.18	1.12	0.32	0.55	2.55	1.38	1.67	0.30
Π_5	0.89	0.89	0.75	0.78	1.13	0.99	0.91	0.25
	$B = 30s$		$B = 30s$		$B = 30s$		$B = 30s$	
Π_6	1.26		1.25		1.27		0.022	
Π_7	1.35		1.347		1.353		0.022	
Π_8	1.22		1.16		1.32		0.12	

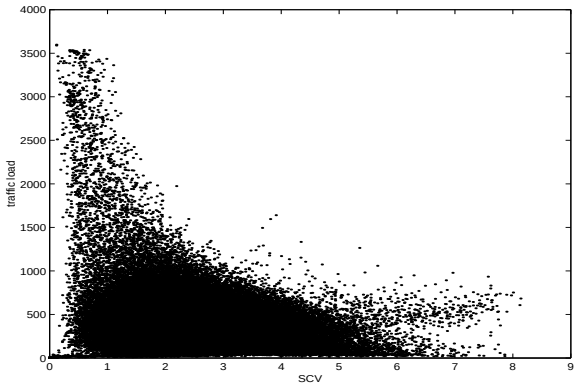


Figure 14: Traffic load vs. SCV.

the adaptive random sampling is likely to offer a higher rate of sampled data reduction in times of high load, while providing the desired degree of accuracy in traffic load estimation. In other words, when a high-speed link is highly utilized, our adaptive sampling technique results in fewer packets to be sampled, thereby reducing the burden (both in terms of processing and storage) on the traffic monitoring and measurement device (whether on-board a router or off-board). And this is achieved *without sacrificing the sampling accuracy*! Lastly, we would like to point out that although our adaptive random sampling technique is designed with the application to load change detection in mind, it can also be applied to other traffic engineering applications.

5 Load Change Detection with Sampled Measurement

Sudden, persistent load changes in network traffic are of great concern to network operators, as they may signal network element failures or anomalous behaviors, and may significantly impact the performance of the network. Hence automatic traffic load change detection is an important aid in network operations and traffic engineering. In this section we present a *non-parametric*, *on-line* change point detection algorithm based on singular value spectrum analysis. The algorithm takes

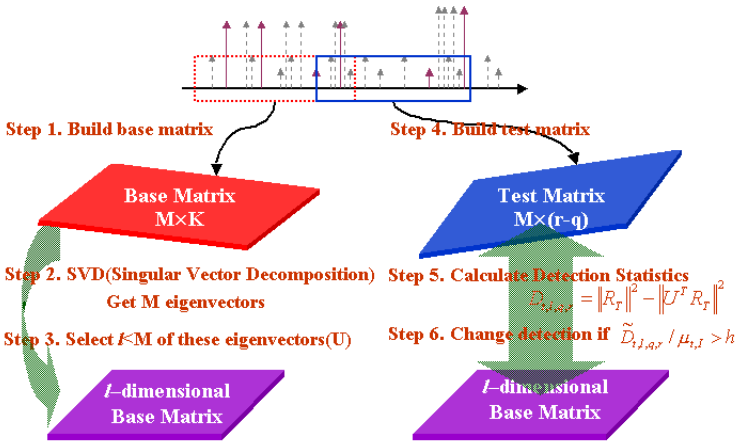


Figure 15: Change detection algorithm.

the time series of estimated traffic loads via sampling, and detects "non-stationarities" (i.e., abrupt changes) in the estimated traffic loads at some (configurable) operational time scale that is larger than the time scale the traffic loads are sampled. We examine the impact of sampling errors on the performance of the load change detection algorithm using real network traces. We also briefly touch on the issues in designing robust load change detection algorithms and the impact of time scale of change.

5.1 Non-parametric On-line Change Point Detection Algorithm

In the problem of traffic load change detection, we assume that the statistics of traffic loads are normally either constant or slowly time-varying; otherwise an *abrupt change* should be recognized. By abrupt changes, we mean changes in characteristics that occur very fast, if not instantly. But before and after the change, the properties are fairly stationary with respect to the time scale of interest. Note that abrupt changes by no means imply changes with large magnitude. Many network management problems are concerned with detection of small changes. Traditional *parametric* techniques involve estimation of certain parameters of the time series such as mean and variance and some presumed distributions (e.g., Gaussian) of the parameters for assessing statistical significance of these estimates. However, such assumptions typically cannot be applied to real network traffic [9]. In particular, a few short bursts (outliers) may greatly distort the estimates. Thus, traditional parametric techniques may not work well in the presence of outliers. A non-parametric algorithm based on singular-spectrum analysis is much more robust, since it efficiently separates noise (outliers) from signal. Since sampling may further introduce or magnify outliers, tolerance of noise is critical in our framework. In addition, amenablity to on-line implementation is another consideration in selecting traffic load change detection algorithms. In our study we employ such a non-parametric change point detection algorithm based on singular spectrum analysis (SSA). The algorithm is developed in [16], which we briefly describe below.

Let $y_1, y_2, \ldots$ be a time series of estimated traffic loads. (Note that the time index t here is in the unit of time block B of sampling.) For each time $t = 1, 2, \ldots$, part of the time series, $y_{t+1}, \ldots, y_{t+N}$, is considered as the "base data." Another sub-series, $y_{t+q+1}, \ldots, y_{t+r+M-1}$, where $q > 0$, is called the "test data." Intuitively, if there is no significant change in the essential traffic signals, the "distance" in statistics between the test data and the base data should stay reasonably

$$Y_B{}^{(t)} = \begin{pmatrix} y_{t+1} & y_{t+2} & \cdots & y_{t+K} \\ y_{t+2} & y_{t+3} & \cdots & v_{t+K+1} \\ \vdots & \vdots & \ddots & \vdots \\ y_{t+M} & y_{t+M+1} & \cdots & y_{t+N} \end{pmatrix} \quad Y_T{}^{(t)} = \begin{pmatrix} y_{t+q+1} & y_{t+q+2} & \cdots & y_{t+r} \\ y_{t+q+2} & y_{t+q+3} & \cdots & y_{t+r+1} \\ \vdots & \vdots & \ddots & \vdots \\ y_{t+q+M} & y_{t+q+M+1} & \cdots & y_{t+r+M-1} \end{pmatrix}$$

Figure 16: Base and test trajectory matrices.

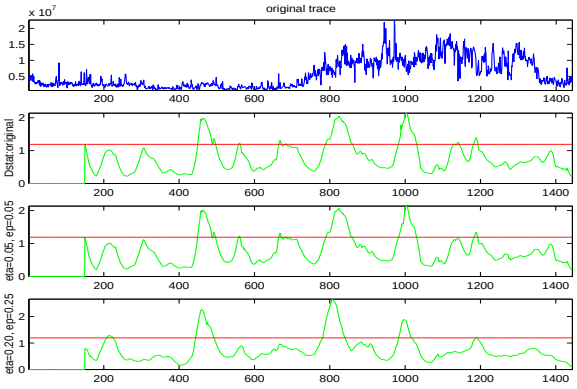


Figure 17: Detection statistics for population and estimated traffic load.

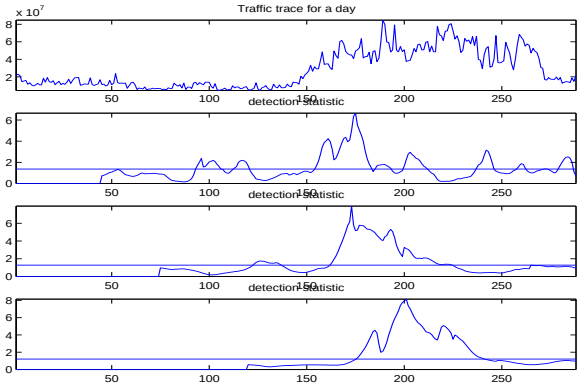


Figure 18: Detection statistics with varying lag parameter.

small. If the "distance" in statistics is larger, it signals an abrupt change. The basic idea of the change point detection algorithm is depicted in Figure 5.1. The steps involved in the change point detection procedure are given below (please refer to [16] for more details).

Let N, M, l, p and q be some integers such that $M \leq N/2$, $0 \leq q < r$ and $N \leq r$. An integer K is set to be $K = N - M + 1$. For each time index $t = 0, 1, \ldots$, compute the following :

1. Build the lag-covariance matrix $R_B{}^{(t)} = \frac{1}{K} Y_B{}^{(t)} (Y_B{}^{(t)})^T$ of the trajectory matrix $Y^{(t)}$ with the base data (as shown in Figure 16).
2. Perform SVD (Singular Value Decomposition) of $R_B^{(t)}$: $R_B^{(t)} = U \Lambda U^T$.
3. Determine a l-dimensional subspace spanned by the first l eigenvectors (U_l) of $R_B^{(t)}$.
4. Similarly, build the lag-covariance matrix $R_T{}^{(t)} = \frac{1}{K} Y_T{}^{(t)} (Y_T{}^{(t)})^T$ of the trajectory matrix with the training data.
5. Compute the detection statistics $\mathcal{D}_{t,l,q,r}$, the sum of the squared Euclidean distances between the vectors $Y_j^{(t)} (j = q+1, \ldots, r)$ and U_l. i.e., $\|R_T\|^2 - \|U_l^T R_T\|^2$.
6. Decide if there is an abrupt change ($\mathcal{D}_{t,l,q,r} > threshold$), and generate an alarm with estimated time $\tau (= t + r + M - 1)$ of change with the detection statistic $\mathcal{D}_{t,l,q,r}$.

5.2 Experiments

The detection statistics $\mathcal{D}_{t,l,q,r}$ holds *asymptotic normality* under the conditions that the window size N and the lag M are sufficiently large [16]; from this result, an asymptotic probability of a change can also be derived. In the results shown in this section, we use a 99% of significance level as

the detection threshold. Figure 17 illustrates the impact of sampling errors on the performance of the load change detection algorithm. The second plot of Figure 17 depicts the detection statistics of the time series of the original traffic loads (with a time block of $B = 60$ seconds), which is shown on the top row. The third and fourth plots are the detection statistics of the estimated traffic loads with the sampling accuracy parameters $\{\eta = 0.05, \varepsilon = 0.05\}$ and $\{\eta = 0.20, \varepsilon = 0.25\}$, respectively. With the sampling accuracy parameters $\{\eta = 0.05, \varepsilon = 0.05\}$, all of the changes detected using the original traffic loads are also detected using the estimated traffic loads. However, with the sampling accuracy parameters $\{\eta = 0.20, \varepsilon = 0.25\}$, one false-alarm (around $200min$)is generated, and two small load changes in the neighborhood of $600min$ and another small load change around $1200min$ are not detected. This evidently tells that bounding estimation errors is critical in traffic load change detection. Note here that the detection algorithm also finds subtle load change points otherwise undetectable via visual inspection by humans without further data processing.

The parameters in the detection algorithm can be tuned to control the time scale of load changes as well as sensitivity (or magnitude) of load changes that are of interest to network operators. Depending on the type of load changes we are looking for, the window size of the base and test matrices, M, and the location and length of the test data (q, r) can be configured accordingly. Observe that if M is too small, then an outlier may be recognized as a structural change. Hence it is recommended that M is chosen to be sufficiently but smaller than the time scale of load changes to be detected so as not to miss out all the changes in the time series of traffic loads. In Figure 18, we show the effect of varying M on the detection statistics. For ease of observation, we use a larger sampling time block ($B = 300$ seconds). The top plot in Figure 18 shows the time series of the original traffic loads. The detection statistics corresponding to $M = 30, 50, 80$ are shown, respectively, in the second, third, and fourth plot. We see that using smaller M detects load changes that occur in a smaller time scale. For example, close visual inspection reveals a small load change in the time scale of 10 units or so (i.e., a duration of about 3000 seconds) before and after the time index 100; a load change of much larger time scale and magnitude occurs around the time index 150, and a few other load changes of smaller time scale occur afterwards. A larger M ignores load changes that occur at the smaller time scale and with smaller magnitude that are otherwise detected by a smaller M. We have further investigated the impact of the control parameters in the detection algorithm on various aspects of load changes (such as speed, time scale of change and sensitivity) for a variety of traffic engineering applications. The initial results are reported in [21]. Due to space limitation, we will not reproduce them here.

6 Conclusions

Network traffic may fluctuate frequently and often unexpectedly for various reasons such as transitions in user behavior and failure of network elements. Timely detection of such changes in traffic is critical for initiating appropriate traffic engineering mechanisms. The performance of a change detection algorithm depends on the accuracy of traffic measurement. But, inspecting *every* packet traversing a link to obtain the *exact* amount of traffic load impairs the processing capacity of a router. Therefore *sampling* techniques that *estimate* traffic accurately with minimal measurement overhead are needed. Static sampling techniques may result in either inaccurate undersampling or unnecessary oversampling. In this paper, we proposed an adaptive random sampling technique that bounds the sampling error to a pre-specified tolerance level while minimizing the number of samples.

We have shown that the minimum number of samples needed to maintain the prescribed accu-

racy is proportional to the squared coefficient of variation (SCV) of packet size distribution. Since we do not have *a priori* knowledge about key traffic parameters – SCV of packet size distribution and the number of packets, these parameters are predicted using AR model. The sampling probability is then determined based on these predicted parameters and thus varied adaptively according to traffic dynamics. From the sampled packets, the traffic load is then estimated. We have also derived a theoretical upper bound on the variance of estimation error which affects the robustness of a change detection algorithm. We have experimented with real traffic traces and demonstrated that the proposed adaptive random sampling is very effective in that it achieves the desired accuracy, while also yielding significant reduction in the fraction of sampled data.

The time series of traffic loads thus estimated are then analyzed using a non-parametric on-line change detection algorithm to find non-stationarities. This algorithm detects changes in the estimated traffic loads at some (configurable) operational time scale that is larger than the time scale at which the traffic loads are estimated. We have investigated the impact of sampling error on the performance of change detection algorithm and illustrated the desirability of bounding estimation error. We believe that our adaptive random sampling technique combined with on-line change detection algorithm can enable intelligent traffic control and engineering in a scalable manner.

References

[1] Nick Duffield, Carsten Lund, and Mikkel Thorup, Charging from Sampled Network Usage, ACM SIGCOMM Internet Measurement Workshop 2001

[2] Cristian Estan and George Varghese, New Directions in Traffic Measurement and Accounting, ACM SIGCOMM Internet Measurement Workshop 2001

[3] Donald A. Berry and Bernard W. Lindgren, "Statistics theory and Methods", 2nd ed., Duxbury Press, ITP, 1996

[4] John M. Gottman, "Time-series analysis", Cambridge University Press, 1981

[5] Nick G. Duffield and Matthias Grossglauser, "Trajectory sampling for direct traffic observation", Proceedings of ACM SIGCOMM 2000 pp271-28.

[6] P. Billingsley, "Convergence of Probability Measures", New York Wisley, 1968 (p.369)

[7] Sampled NetFlow. http://www.cisco.com/univercd/cc/tc/doc/product/software/ios120/120newft

[8] PMA Traces Archive http://moat.nlanr.net utilization

[9] Walter Willinger, Murad Taqqu, and Ashok Erramilli, "A Bibliographical Guide to Self-Similar Traffic and Performance Modeling for Modern High-Speed Networks Stochastic Networks: Theory and Applications", Royal Statistical Society Lecture Notes Series, Vol. 4, Oxford University Press, 1996.

[10] Kimberly C. Claffy, George C. Polyzos and Hans-Werner Braun, Application of sampling methodologies to network traffic characterization, in Proceedings ACM SIGCOMM'93, San Francisco, CA, September 13–17, 1993.

[11] Aimin Sang, S. Q. Li, A Predictability Analysis of Network Traffic, in Proceedings of IEEE INFOCOM'2000.

[12] R.E. Moore, Problem detection, isolation and notification in systems network architecture, in *Proc. of IEEE Infocom'86* 1986.

[13] The Surveyor Project Advanced Networks, http://www.advanced.org/surveyor and http://betelgeuse.advanced.org/csg-ippm/

[14] John R. Wolberg, "Prediction Analysis", Princeton, N.J., B. Van Nostrand, 1967

[15] Michele Basseville and Igor V. Nikiforov, "Detection of Abrupt Changes: theory and Application", Prentice-Hall, Inc. Englewood Cliffs, N.J., ISBN 0-13-126780-9, April 1993

[16] V. Moskvina and A. Zhigljavsky. "Change-point detection algorithm based on the singular-spectrum analysis, Detection", School of Mathematics, Cardiff University, Senghennydd Road, Cardiff, CF24 4YH, UK, *Preprint*

[17] V. Moskvina, " Distribution of random quadratic forms arising in singular-spectrum analysis", Mathematical Communications, 5, 161-171, 2000

[18] C. R. Rao, "Sampling Techniques" 2nd ed., N.Y., Wiley. 1973

[19] A.K. Bera and C.M. Jarque, "An efficient large-sample test for normality of observations and regression residuals", *Working Papers in Economics and Econometrics*, 40, Australian National University, 1981

[20] K. Thompson, G. Miller and R. Wilder, "Wide-Area Iternet Traffic Patterns and Characteristics", *IEEE Network* Nov/Dec. 1997

[21] B. Choi, J. Park, Z. Zhang, "On Abrupt Traffic Change Detection", *in preparation*

A Appendix

In this appendix we sketch the proofs of Lemma 1 and Theorem 2. For simplicty of notation, we drop the subscript k in the notation.

Proof [: of Lemma 1] Note first that

$$\lim_{m\to\infty} \frac{m}{\hat{m}} = \lim_{m\to\infty} \frac{m}{m+e_m} = 1.$$

Since $S^s \to S$ as $n \to \infty$, $\hat{n} = z_p S^s \to z_p S = n$. From $\tilde{n}_k = \hat{n} \cdot \frac{m}{\hat{m}}$, we have $\lim_{n\to\infty} \frac{\tilde{n}}{n} = 1$. ∎

Proof [: Theorem 2]

$$\begin{aligned}
\tilde{V} &= \frac{m}{\tilde{n}} \sum_{i=1}^{\tilde{n}} X_i \\
&= \frac{m}{\tilde{n}} (\tilde{n}\mu^s + \sigma^s \sqrt{n} Y + o(\sqrt{n})) \\
&= \hat{V} + \frac{(\sigma^s \sqrt{n} Y + o(\sqrt{n}))\hat{m}}{\hat{n}} \\
&\approx \hat{V} + \frac{\hat{m}}{z_p \hat{S}^s} \sqrt{S^s} \mu^s \sqrt{z_p S} Y \\
&= \hat{V} + \frac{\hat{m}\mu^s}{\sqrt{z_p}} \cdot \frac{\sqrt{S(S+Z)}}{S+Z+e^s} Y
\end{aligned} \tag{17}$$

where, Z is a normal random variable with mean 0, and Y is standard normal random variable. Note that

$$\frac{\sqrt{S(S+Z)}}{S+Z+e^s} \leq \frac{S+Z/2}{S+Z+e^s} \leq \frac{S+Z/2}{S+Z} \leq 1$$

Since $\left|\frac{\hat{V}-V}{V}\right| < \varepsilon$ with probability $1-\eta$, the following holds with probability $1-\eta$.

$$\left|\tilde{V} - V\right| < V\varepsilon + \frac{V}{\sqrt{z_p}}(1+\varepsilon)Y + \frac{e_m \mu^s}{\sqrt{z_p}} Y \tag{18}$$

Therefore, the relative error is given by

$$\begin{aligned}
\left|\frac{\tilde{V} - V}{V}\right| &< \varepsilon + \frac{1}{\sqrt{z_p}}(1+\varepsilon)Y + o(\frac{1}{m}) \\
&\approx \varepsilon + \frac{1}{\sqrt{z_p}}(1+\varepsilon)Y
\end{aligned}$$

∎