Inferring original traffic pattern from sampled flow statistics

Tatsuya Mori Ryoich Kawahara Noriaki Kamiyama Shigeaki Harada
NTT Service Integration Laboratories, NTT Corporation
3-9-11 Midoricho Musashino-shi Tokyo, Japan
tatsuya@nttlabs.com, {kawahara.ryoichi, noriaki.kamiyama, harada.shigeaki}@lab.ntt.co.jp

Abstract

Packet sampling has become a practical and indispensable means to measure flow statistics. Recent studies have demonstrated that analyzing traffic patterns is crucial in detecting network anomalies. We may not be able to infer the original traffic patterns correctly from the sampled flow statistics because sampling process wipes out a lot of information about small flows, which play a vital role in determining the characteristics of traffic patterns. In this paper, we first show an example of how the sampling process wipes out the original statistics using measured data. Then, we show empirical examples indicating that the original traffic pattern cannot be inferred correctly even if we use a statistical inference method for incomplete data, i.e., the EM algorithm, for sampled flow statistics. Finally, we show that additional information about the original flow statistics, the number of unsampled flows, is helpful in tracking the change in original traffic patterns using sampled flow statistics.

1. Introduction

With the recent growth in link speed, packet sampling has attracted much attention as a scalable means to measure network traffic from both industrial and research communities [1, 6]. The technique has been standardized in the IETF psamp working group [2], and most major vendors have embedded packet sampling functions into their router products. Most large ISPs, which are operating high-bandwidth links such as OC192, are monitoring their network using packet sampling techniques.

Many recent studies have demonstrated that analyzing traffic patterns is crucial in detecting network anomalies [4, 7]. For example, a sharp increase in the number of small flows, e.g., those with $1-3$ packets, may be related to an anomalous event such as SYN flooding or a worm outbreak. Such events cannot be characterized by conventional volume-based statistics such as byte or packet counts. Lakhina et al. [4] showed that the change in traffic pattern can be successfully characterized by information entropy, which can effectively express the concentration or dispersion of the distributions of observed random variables. However, we may not able to infer the original traffic patterns correctly from the sampled flow statistics because the sampling process wipes out a lot of information about small flows, which play a vital role in determining the characteristics of traffic patterns (see [3] for a detailed analysis). Therefore, studying how the sampling affects the observed traffic patterns is meaningful.

In this paper, first, we demonstrate that packet sampling with a low sampling frequency, e.g, $f = 10^{-3}$, which is a commonly used parameter setting in backbone links, causes the change in traffic patterns to be undetected. Then, we show experimental examples in which even if we use the statistical inference technique for the incomplete data, i.e., the EM algorithm, the original traffic pattern cannot be reconstructed correctly. Finally, we show that the maximum likelihood estimation with additional information about the original flow statistics, the number of unsampled flows, is effective in tracking the change in original traffic patterns.

2. Traffic pattern and packet sampling

This section shows how packet sampling affects observed traffic patterns. Throughout this paper, we use the publicly available packet traces obtained from [5]. The traces used in this paper were measured on one of the International backbone links of the WIDE project January 7, 2005 from 19:00 to 23:00. The measurement bin was set to 5 minutes. Several time series of original and sampled flow statistics with the sampling frequencies of $f \in \left\{10^{-2}, 10^{-3}\right\}$ are shown in Fig. 1. Here, we emulated random packet sampling. We focus on (1) the number of observed (sampled) flows, (2) the ratio of one-packet flows (OPF ratio), and (3) entropy when analyzing flow statistics. Here, the OPF ratio is the ratio of the number of flows that comprise exactly one packet divided by the total number of flows, and the entropy s is defined as $s = -\sum_i p(i) \log p(i)$, where $p(i)$ is the probability that a flow has i packets.

In the original flow statistics, we see that the number of flows has a sharp spike around 21:20. Through the detailed analysis of measured data, we found that the spike is due to a severe SYN flooding attack against several destination addresses from a number of possibly spoofed source addresses. We can also see that the increase/decrease in

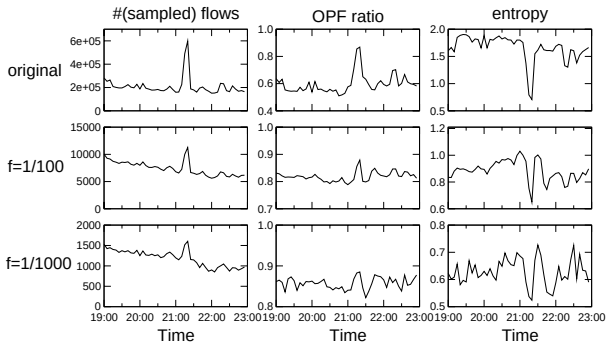

Figure 1. Time series of original and sampled flow statistics.

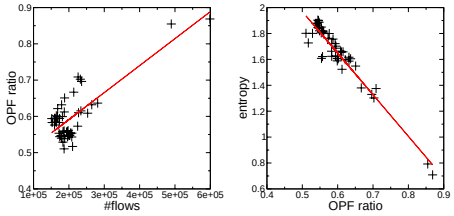

Figure 2. Correlation among original flow statistics.

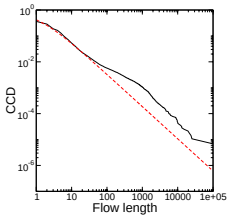

Figure 3. Flow length distribution approximated using Pareto distribution.

the OPF ratio roughly corresponds to the decrease/increase in entropy, which means that the concentration of the OPF decreases the entropy. As shown in Fig. 2, the number of flows and OPF ratio are not well correlated while OPF ratio and entropy are fairly well correlated negatively. Actually, the correlation coefficient between the number of flows and OPF ratio is 0.795 while that between the OPF ratio and entropy is -0.958. Thus, entropy can effectively reflect the change in flow mix, i.e., increase in OPF ratio in this case, while the number of flows cannot always correctly reflect the change in the flow mix. The increase in OPF is related to some anomalous events such as SYN flooding, so network scanning and port scanning, using entropy as a measure of a traffic pattern are quite useful.

On the other hand, as the sampling frequency decreases, we see that the changes in these statistics become totally undetectable because most OPFs are not sampled. This observation empirically suggests that detecting network anomalies from sampled flow statistics is a difficult task. In the next section, we describe how to infer original flow statistics from the sampled flow statistics.

3. Inferring original flow length distribution

3.1 Modeling flow length distributions

The flow length distribution in the Internet is well known to be heavy-tailed, which is considered as one of the *invariant* characteristics of the Internet [6]. Here, flow length means the number of packets in a flow. The Pareto distribution is known to be the simplest model for characterizing heavy-tailed distributions. Thus, we adopt the Pareto distribution for modeling flow-length distributions. Adopting more complex distribution models or nonparametric approaches is also possible [1]. However, as we shall see below, the modeling of flow-length distributions with the Pareto distribution is quite effective in characterizing traffic patterns in terms of entropy.

The probability density function of the Pareto distribution is given by $f(x) = \theta a^\theta / x^{\theta+1}$, where $x \geq a$. We model the flow length X with the Pareto distribution. The random variable X takes positive discrete values $1, 2, 3, \ldots$, so we set the scale parameter to $a = 1$ and discretize the distribution with the following probability mass function $p(k;\theta)$.

$$p(k;\theta) = \Pr[X = k] = \int_k^{k+1} f(x)dx = k^{-\theta} - (k+1)^{-\theta} \quad (1)$$

The ML estimation of parameter θ can be obtained by numerically solving the following log-likelihood equation,

$$\sum_{k=1}^{x_{\max}} N_k \frac{-k^{-\theta}\log k + (k+1)^{-\theta}\log(k+1)}{k^{-\theta} - (k+1)^{-\theta}} = 0,$$

where N_k is the number of flows that comprise k packets in the original flows and $x_{\max}$ is the maximum flow length in the original flows.

The log-log complementary distribution (LLCD) plots of the empirical distribution of the original flow lengths and its Pareto approximation (dashed line) for one of the traces are shown in Fig. 3. We observe that the Pareto model fits the empirical distribution fairly well especially in the range of small flow lengths. The comparison between the actual entropy and the estimated entropy obtained with the Pareto approximation for all the traces is shown in Fig. 4. We can see that the actual entropy and the estimated entropy agree quite well; the correlation coefficient was 0.999. This comes from the fact that entropy is strongly affected by the flow lengths with high probabilities, i.e., small flow lengths, which are well approximated by the Pareto distribution.

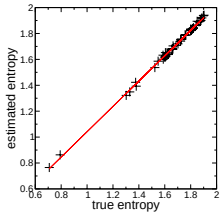

Figure 4. Actual entropy vs. the estimated entropy using Pareto approximation.

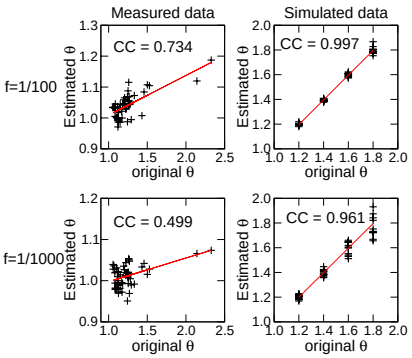

Figure 5. Original θ vs. estimated $\widehat{\theta}$ using EM algorithm.

3.2 Inferring with the EM algorithm

We aim to infer the original flow distribution from the incomplete observation, i.e., sampled flow statistics. Suppose N packets appear during a certain time period, and each of them is sampled independently with probability f. In other words, we consider random sampling from a population of N packets with sampling frequency f. We define X_j and Y_j $(j = 1, 2, \ldots, M)$ as the original and sampled flow lengths of the j-th flow, respectively. By definition, $N = X_1 + X_2 + \cdots + X_M$. Let n_i $(i = 0, 1, \ldots, y_{\max})$ denote the number of flows whose sampled flow length is equal to i, where $y_{\max}$ denotes the maximal sampled flow length. We consider random sampling, so the conditional probability that a sampled flow has i packets given that the flow originally has k packets is given by the binomial distribution, $q(i \mid k) = \binom{i}{k} f^i (1-f)^{k-i}$. Thus, the probability that a sampled flow has i packets is given by $r(i;\theta) = \sum_{k=i}^{N} q(i \mid k)\, p(k;\theta)$, where $p(k;\theta)$ is given by Eq. (1). We estimate parameter θ from the observed statistics $n_1, n_2, \ldots, n_{y_{\max}}$. Note that we cannot observe n_0, the number of flows with zero packets sampled. The complete-data log likelihood for sampled flow statistics is given by

$$\log L_c(\theta) = \log \prod_{i=0}^{y_{\max}} r(i;\theta)^{n_i} = \sum_{i=0}^{y_{\max}} n_i \log r(i;\theta). \qquad (2)$$

E-step: Let $\theta^{(0)}$ be the initially specified value for θ. The conditional expectation of $\log L_c(\theta)$ given $n_1, \ldots, n_{y_{\max}}$ using $\theta^{(0)}$ can be written as $Q(\theta;\theta^{(0)}) = E_{\theta^{(0)}}\{\log L_c(\theta) \mid n_1, \ldots, n_m\}$. We have $\log L_c(\theta)$ as a linear function of the unobservable data n_0, so the E-step is established by replacing n_0 with its current conditional expectations given the observed data. Let the conditional expectations of n_0 be $n_0^{(0)}$. We estimate $n_0^{(0)}$ using the odds ratio, i.e.,

$$n_0^{(0)} = E_{\theta^{(0)}}(n_0 \mid n_1, \ldots, n_{y_{\max}}) = \frac{r(0;\theta^{(0)})}{1 - r(0;\theta^{(0)})} \sum_{i=1}^{y_{\max}} n_i. \qquad (3)$$

M-step: The M-step is undertaken on the first iteration by choosing $\theta = \theta^{(1)}$ that maximizes $Q(\theta;\theta^{(0)})$. Such a value can be obtained by numerically solving $\frac{\partial Q(\theta;\theta^{(0)})}{\partial \theta} = 0$. Then, the new $\theta^{(1)}$ is substituted into the right-hand side of Eq. (3) to produce an updated value of $n_0^{(1)}$. In the same manner, we repeat the E- and M-steps alternately until the estimated parameter $\theta^{(k)}$ on the kth iteration satisfies $\left|\theta^{(k)} - \theta^{(k-1)}\right| < 10^{-4}$.

3.3 Validation of accuracy

We apply the inference technique proposed in the previous section to both measured and simulated data. The simulated data produced by the ideal discrete Pareto distribution with the shape parameter of $\theta \in \{1.2, 1.4, 1.6, 1.8\}$, where we produced 11 distinct data sets for each shape parameter using different random seeds to generate the data. In total, we examined 48 measured data sets and 44 simulated data sets. For both data sets, we emulated the random packet-sampling process with sampling frequencies of $f \in \left\{10^{-2}, 10^{-3}\right\}$. The results are shown in Fig. 5, where CC means the correlation coefficient. For the measured data, we see that the estimation is not so good and the estimated values are underestimated compared to the actual values. This trend becomes more severe as the sampling frequency decreases. We conjecture that the reason why the parameters are not correctly estimated and underestimated is because of the approximation of the Pareto distribution. As shown in Fig. 3, the approximation is not so good for large flow lengths. Moreover, most sampled-flow statistics come from large flows. Therefore, a possible approach to improve the estimation accuracy may be to use the information about the *unsampled* statistics, as we will see below. On the other hand, for simulated data, the estimations are fairly good for $f = 10^{-2}$. However, the estimations deviate for $f = 10^{-3}$, especially for large θ.

4. Use of number of unsampled flows

From the sampled-flow statistics, we cannot observe n_0, which is the number of flows that are not sampled. In this

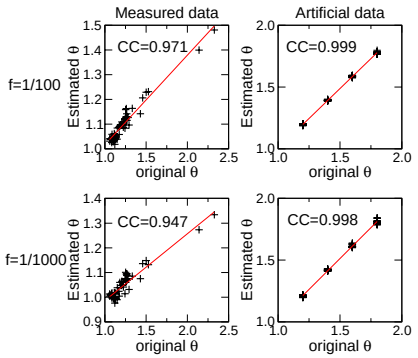

Figure 6. Original θ vs. the estimated $\widehat{\theta}$ using presented method.

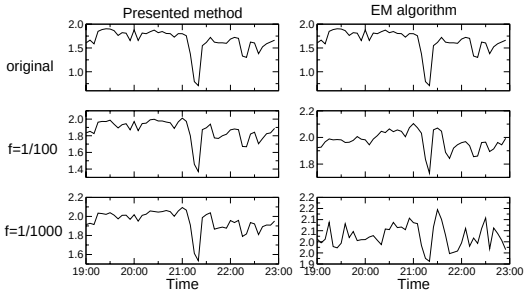

Figure 7. Estimated entropy by presented method (left) and EM algorithm (right). Top is actual entropy.

section, we show an empirical study of how the information about n_0 improves the inference. Meanwhile, to obtain such statistics, we have two possible solutions. One is to infer the number of original flows from the statistics of TCP flags that are included in common flow records such as NetFlow. This idea was proposed by Duffield et al. in [1]. That is, the number of TCP flows can be inferred using the number of sampled flows whose SYN flag is set. Assuming that most flows on the Internet are TCP flows and each flow has one SYN packet, the number of original flows can be inferred as $\widehat{M} = f^{-1}m_s$, where m_s is the number of sampled flows with the SYN flag set. The disadvantage of this approach is that non-TCP flows such as UDP and ICMP are not considered. Another approach is to measure the number of flows directly. We can adopt several techniques to achieve such measurement that can be achieved in high-speed networks with very small amount of memory, e.g., probabilistic counting or bloom filter. The disadvantage of this approach is that the addition of new measurement functions on routers is required.

We assume that we can obtain n_0 through the above approaches. Then, we find the parameter θ that maximizes the complete-data log likelihood defined in Eq. (2). That is, we numerically solve the equation, $\frac{\partial \log L_c(\theta)}{\partial \theta} = 0$. Here, we used the actual value of n_0 for each data set. The results are shown in Fig. 6. Although the parameters are underestimated again, the correlations with the original parameters are significantly improved. We conjecture that the reason why the presented method, i.e., the MLE with the information about n_0, underestimates parameters is because we do not have any information about the flow mix of the unsampled data, which cannot be obtained unless we know the original distributions. To see the usefulness of the estimated parameters, we calculate entropy using the estimated parameters (see Fig. 7). For comparison, we also plot the original entropy. We see that the presented method tracks the change in the original traffic pattern well while the estimations using the EM algorithm fail.

5. Conclusion

We demonstrated that packet sampling causes the change in traffic pattern to become undetected and the original traffic pattern cannot always be reconstructed even if we use the EM algorithm for the sampled-flow statistics. We also showed that additional information about the original flow statistics, the number of unsampled flows, is effective in tracking the change in original traffic patterns. We believe that this approach is promising in establishing a scalable network operation scheme that can also detect network anomalies.

Acknowledgments We are grateful to Professor Tetsuya Takine for providing the idea of the parametric approach to infer flow statistics. This work was supported by the Ministry of Internal Affairs and Communications, Japan.

References

[1] N. G. Duffield, C. Lund, and M. Thorup. Estimating flow distributions from sampled flow statistics. In *Proc. ACM SIGCOMM*, pages 325–336, August 2003.

[2] IETF Packet Sampling (psamp) Working Group. http://www.ietf.org/html.charters/psamp-charter.html.

[3] R. Kawahara, T. Mori, N. Kamiyama, S. Harada, and S. Asano. A study on detecting network anomalies using sampled flow statistics. Proc. SAINT Workshop.

[4] A. Lakhina, M. Crovella, and C. Diot. Mining anomalies using traffic feature distributions. In *Proc. ACM SIGCOMM*, pages 217–228, September 2005.

[5] Measurement and Analysis on the WIDE Internet. http://www.wide.ad.jp/wg/mawi/.

[6] T. Mori, M. Uchida, R. Kawahara, J. Pan, and S. Goto. Identifying elephant flows through periodically sampled packets. In *Proc. ACM IMC*, pages 115–120, October 2004.

[7] K. Xu, Z. L. Zhang, and S. Bhattacharyya. Profiling internet backbone traffic: Behavior models and applications. In *Proc. ACM SIGCOMM*, pages 169–180, September 2005.