Lightweight Link Dimensioning using sFlow Sampling

R. de O. Schmidt[*], R. Sadre[†], A. Sperotto[*] and A. Pras[*]

[*]University of Twente, The Netherlands
{r.schmidt,a.sperotto,a.pras}@utwente.nl
[†]Aalborg University, Denmark
rsadre@cs.aau.dk

Abstract—**Operators use link dimensioning to provision network links. In practice, traffic averages are obtained via SNMP are used to roughly estimate required capacity. More accurate solutions often require traffic statistics easily obtained from packet captures, e.g. variance. However, packet capturing may not be trivial in high-speed links. Aiming scalability, operators often deploy packet sampling on monitoring, but little is known how it affects link dimensioning. In this paper we assess the feasibility of lightweight link dimensioning using *sFlow*, which is a widely-deployed traffic monitoring tool. We implement sFlow sampling algorithm and use a previously proposed and validated dimensioning formula that needs traffic variance. We validate our approach using packet captures from real networks. Results show that the proposed procedure is successful for a range of sampling rates and that, due to randomness of sampling algorithm, the error introduced by scaling the traffic variance yields more conservative results that cope with short-term traffic fluctuations.**

Index Terms—**Link dimensioning; packet sampling; sFlow.**

I. INTRODUCTION

Link dimensioning is often used by network operators aiming at optimal resources allocation in the network infrastructure or for network planning. Commonly, network operators use traffic averages obtained by pulling SNMP MIBs (e.g., octet counter), with graphical support provided by tools such as MRTG (*Multi Router Traffic Grapher*). These approaches, often referred to as *rules of thumb*, simply add to the traffic average a fixed amount of bandwidth as a safety margin. This margin may depend on other factors such as time of the day. Several alternatives to these simple rules of thumb have been proposed in the past. Most of these require packet-level measurements. Traffic statistics are computed at packet level and applied to dimensioning formulas. For instance, in [1], [2] the authors propose and validate a dimensioning formula that requires traffic mean and traffic variance. Both parameters can easily be obtained from packet traffic measurements. However, due to increasing traffic rates, full packet capturing in high-speed links may not be trivial. To avoid measurements overload and still have packet-level granularity, network operators commonly deploy packet sampling within the traffic monitoring process. *sFlow*, for example, is a widely deployed tool that has the goal of enabling packet-level traffic monitoring in high-speed switched networks. sFlow provides sampled packet measurements that could be used for link dimensioning. However, little is known about the impact of sampled data on link dimensioning procedures.

Contribution. In this paper we assess the feasibility of lightweight link dimensioning using sFlow sampling algorithm. We adopt the link dimensioning formula proposed in [1], [2]. Such formula requires traffic statistics that may be affected by packet sampling, namely mean rate and traffic variance. We assess the impact of various sampling rates on estimations of required link capacity. To validate our experiments, we apply the sFlow sampling algorithm as found in typically deployed sFlow tools to real network traffic traces captured in several different locations around the globe. We show that even with sampled data and simple procedures for scaling traffic average and variance, the results from the dimensioning formula are accurate when compared to an empirically defined ground truth. It is important to highlight that the evaluation and comparison of different packet sampling strategies is not the focus of this paper and, hence, we only implement the sampling approach as defined in [3].

Related work. To the best of our knowledge, works on assessing the impact of sFlow sampling on link dimensioning have not been done yet. However, previous works such as [6] have proposed new sampling approaches, namely adaptive sampling, in which they can estimate traffic load and variance from sampled traces. Adaptive packet sampling was also studied in [7] in the context of flow-level traffic measurements. Our work differs from the previous ones in that we do not aim at proposing a new sampling strategy, but to study the potential of having a lightweight link dimensioning procedure that uses sampled traffic data provided by a widely deployed monitoring tool. We believe that gaining this understanding can positively affect the deployment of advanced link dimensioning procedures in operational networks. In addition, we validate our study on a large and heterogeneous measurements dataset consisted of real network traffic traces.

II. SFLOW SAMPLING

sFlow [3] is a monitoring technology that uses packet filtering and sampling to provide scalable packet-based monitoring in high-speed networks. The monitoring architecture of sFlow consists of agents embedded in switches and routers and a centralized collector. In the context of this work, we focus on the sFlow packet sampling algorithm that is located within

the sFlow agents. According to [3], in sFlow a flow is defined as the set of all packets that are received by an interface, go through the switch or router and are sent to another interface.

In the context of link dimensioning we are interested in the whole traffic aggregate and, therefore, we assume that all observed packets undergo the sampling procedure (i.e., no packet filtering). Although different sampling algorithms can be used, for this paper we consider the random sampling strategy described in the documentation available at InMon website[1] and implemented by well-known sFlow tools, such as `pmacct`[2]. In such algorithm, the decision of sampling a packet is based on a randomly generated counter such that in average 1 in N packets are sampled. This counter tells the algorithm how many packets to skip before sampling one. The counter is therefore progressively decremented for every received packet until it triggers the sampling of a packet. Typically, the random number generator yields uniformly distributed numbers and is seeded with the system's current time. Note that implementations of the sampling algorithm may vary from vendor to vendor. The study of the impact of other sampling techniques on link dimensioning is planned as future work.

III. LINK DIMENSIONING

The link dimensioning approach used in this paper was originally proposed and validated for traffic measurements *without* sampling in [1], [2]. Aiming at "link transparency", this approach provides dimensioning in which users almost never perceive network performance degradations due to lack of bandwidth. To statistically assure transparency to users, the provided link capacity C should satisfy $\mathbb{P}\{A(T) \geq CT\} \leq \varepsilon$, where $A(T)$ denotes the total amount of traffic arriving in intervals of length T, and ε indicates the probability that the traffic rate $A(T)/T$ is exceeding C at the timescale T. In [1], [2] a bandwidth provisioning formula is provided that requires that traffic aggregates $A(T)$, at timescale T, are *Normal distributed* and *stationary*. The link capacity $C(T, \varepsilon)$ needed to satisfy the condition above can be calculated by:

$$C(T, \varepsilon) = \rho + \frac{1}{T}\sqrt{-2\log(\varepsilon) \cdot \upsilon(T)} \quad , \qquad (1)$$

where the mean traffic rate ρ is increased by a term that can be seen as a "safety margin" depending on the variance $\upsilon(T)$ of $A(T)$.

Relying on the variance $\upsilon(T)$ this bandwidth provisioning formula is able to take into account the impact of possible traffic bursts on the required link capacity. In addition, it is very flexible: network operators can choose T and ε according to the QoS that they want to provide to their customers. For example, while larger T (i.e., around 1s) would be enough to provide good quality of experience to users on web browsing, shorter T (i.e., milliseconds scale) should be chosen if real time applications are predominant in the network. The value for ε should be chosen in accordance to the desired QoS.

Essentially, the lower ε the more importance is given to traffic bursts of size T when computing the required link capacity.

Eq. (1) requires that a good estimation of the mean traffic ρ and the variance $\upsilon(T)$ is available. In order to apply the equation to sampled traffic, we propose the following simple procedure to estimate the original mean and the variance of the original traffic (before sampling). Let $L_i(T)$ be the amount of sampled traffic (in bytes) observed in time interval i of length T. The original amount of traffic $A_{i,est}(T)$ in that interval can be estimated by:

$$A_{i,est}(T) = r \cdot L_i(T) \quad ,$$

where r is the inverse of the sampling rate (e.g., $r = 100$ for $1:100$ sampling). The estimated mean ρ_{est} and variance $\upsilon_{est}(T)$ are given by, respectively:

$$\frac{r}{nT}\sum_{i=1}^{n} L_i(T) \quad \text{and} \quad \frac{r^2}{n-1}\sum_{i=1}^{n}(L_i(T) - \rho_s)^2 \quad ,$$

where n is the number of monitored intervals of duration T and ρ_s is the mean traffic rate of the sampled traffic. As can be seen, the mean and variance of the sampled traffic are scaled by a factor of r and r^2, respectively.

It should be noted that, while ρ_{est} is an unbiased estimator of the mean traffic rate ρ, the variance may be overestimated especially for small T and large r because the additional variance introduced by the sampling process is not taken into account. However, the scaling by r, respectively r^2, is easy to implement and does not require any modification in the equipment nor any parameter tuning by the user. The impact of the estimation error on the performance of the link dimensioning formula is further studied in this paper. It might be possible to obtain better estimators of traffic statistics. For example, we could make use of the *sample pool* (i.e., packet counter) available in sFlow datagrams to calculate the precise sampling rate on every datagram. Such investigation is, however, envisioned as future work.

IV. EXPERIMENTS

A. Measurements Dataset

The dataset used in this paper consists of IP traffic captured using tools such as `tcpdump`. Packet-level measurements allow us to validate the experimental results against an empirically defined ground truth. The dataset comprises 284 15-minute traces[3], totaling 71 hours of captures. These captures were done in 6 different locations, from 2011 to 2012, and account for a total of more than 11 billion packets.

In location A (2011), an aggregate link 2×1 Gb/s was measured during 24 consecutive hours. This link connects a building to the gateway of a university. Measurements of location B (2012) and C (2012) took place at the gateway of two universities. In B a 10 Gb/s link was measured for 15 minutes every hour during 24 consecutive hours. The link comprises all incoming and outgoing traffic from the university

---

[1] http://www.inmon.com
[2] http://www.pmacct.net

[3] Trace duration of 15 minutes has been chosen in accordance with [1], [2]. Longer time periods are generally not stationary due to the diurnal pattern.
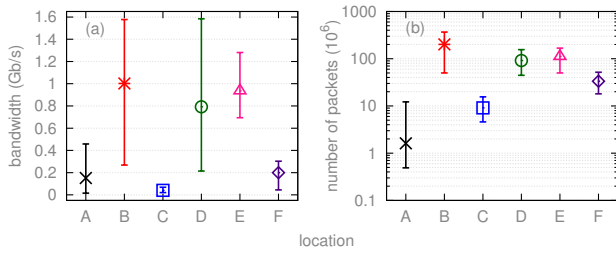
Fig. 1. (a) mean traffic; (b) average number of packets.

Fig. 2. CDF of γ for all traces at $T = 1$s: (a) complete and (b) $1 : 100$.

including student residences. In C an aggregate link (155 Mb/s and 40 Mb/s) was measured. C's traces are captures of the first 15 minutes of every full hour during several days. Traces from locations D (2011) and E (2011–2012) come from the CAIDA public repository[4]. These are 1-hour long traces measured in different days. Two 10 Gb/s links, interconnecting two U.S. cities, were measured at each location. Finally, traces from location F (2012) come from the MAWI public repository. F consists of traffic captures in a trans-pacific link. No additional information on link capacity and load is provided by MAWI[5]. For all captures directly performed by us (i.e., locations A, B and C), no packet losses were observed. From the CAIDA website we know that, for one link of the location D's pair, packet losses are likely to happen (but they do not keep record of such losses). For traces from location F, no information on packet loss is provided by MAWI.

Due to the nature of measured links, mean traffic is not expected to be constant over the whole measurement period. Fig. 1a shows the average, minimum and maximum traffic rate per 15 minutes for each location. Locations with higher-capacity links are the ones in which traffic varies most. In the case of the 24-hour measurements from A and B, differences between minimum and maximum rates are due to traffic dissimilarities in diurnal and overnight periods. Fig. 1b shows the average number of packets per 15-minute trace for each location. From this figure, one can infer, for each location, the average amount of packets after applying packet sampling techniques with different rates.

In this work we focus on an offline sFlow operation due to the fact that we need the raw packet data, i.e., the complete packet traces, in the validation of the experimental results. The sampling is, therefore, performed offline by applying the algorithm described in Sec. II.

B. Traffic Gaussianity

Gaussianity of traffic is a key requirement of Eq. (1) and, hence, it is an important part of the validation procedure to assure that sampled traffic is still Gaussian. Let T be the timescale of traffic aggregation and $L_1(T), \ldots, L_n(T)$ the amount of traffic observed in time periods $1, 2, \ldots, n$ of length T. For any $T > 0$, we want to know if $L(T)$ is Normal

[4]The CAIDA UCSD Anonymized Internet Traces 2011 and 2012. Available at http://www.caida.org/
[5]Information on the link capacity provided on http://mawi.wide.ad.jp is not consistent with the throughput observed in the traces.
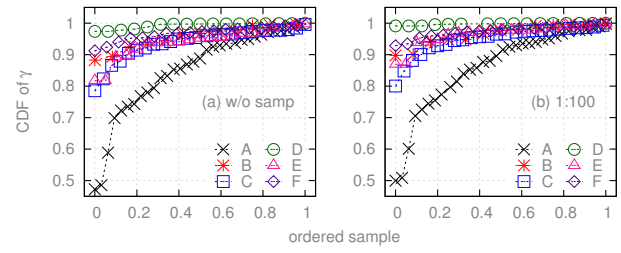
distributed, i.e., whether $L(T) \sim N(\mu, \sigma^2)$. To quantify Gaussianity fit we use the *linear correlation coefficient* [9], which is defined by:

$$\gamma(x, y) = \frac{\sum_{i=1}^{n}(x_i - \overline{x})(y_i - \overline{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \overline{x})^2 \sum_{i=1}^{n}(y_i - \overline{y})^2}} \quad ,$$

where x is the inverse of the normal cumulative distribution function of the sample and y is the ordered sample. One can find more details on how γ is defined in [9]. It is important to know that $\gamma \geq 0.9$ supports the hypothesis that the underlying distribution is normal. A sample is "perfectly Gaussian" when $\gamma = 1$. Fig. 2 shows the CDF of γ for all traces per location. Around 80% of all traces in our dataset is at least "fairly Gaussian" even sampled at $1 : 100$. Most of the traces from A that have $\gamma < 0.9$ were captured overnight when less users are active resulting in reduced Gaussian fit due to smaller traffic aggregate. The main take away of this figure is that the Gaussian property of traffic remains even after sampling with rate $1 : 100$. Clearly, by increasing the sampling rate the Gaussian character diminishes. Fig. 2 only shows γ for $T = 1$s. However, we have assessed Gaussian character of our dataset and it persists over different T. A thorough study on Gaussian properties of our dataset can be found in [8].

C. Impact of Sampling on Link Dimensioning

We assess impact of sampling on the accuracy of the link dimensioning procedure by comparing estimations from Eq. (1) using the complete data and using sampled data scaled using the method described in Sec. III. Fig. 3 shows, for an example trace per location, the average, minimum and maximum relative deviation of the estimated required capacity, computed over 10 runs of sFlow sampling, from the required capacity computed over the complete trace. All calculations were done with $\varepsilon = 1\%$ and at different timescales. The main take away of Fig. 3 is that at shorter T the difference between results from sampled and complete traces is likely to be higher than for larger T. For example, the average difference of around 13% for the trace from location E at $T = 1$ms drops to 1% at $T = 100$ms and becomes around 0.14% at $T = 1$s. The difference at larger T can be considered negligible. The worst result at $T = 1$s is obtained for location C, where the maximum value estimated from the sampled trace was around 0.55% higher than the estimation from the complete trace. The higher overestimation at shorter T might be caused by our simplistic scaling approach. While at larger T actual traffic
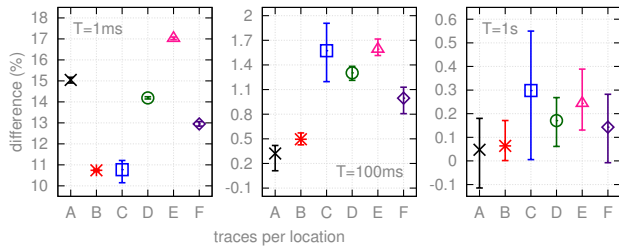
Fig. 3. $C_{1:\,10}(T,\varepsilon)$ as compared to $C(T,\varepsilon)$; results from 10 runs of sampling; one example trace per location; $\varepsilon = 1\%$.

fluctuations were faithfully reproduced by the used scaling approach, at shorter T, such as $T = 1ms$, traffic peaks are mistakenly created due to the random nature of the sampling. These peaks resulted from scaling several successive packets that were sampled in a very short period of time.

Given that the variation of the estimations over the 10 runs is very small, for the results in Fig. 4 were obtained from a single sampling run. Fig. 4 shows the average obtained error of required capacity estimation. This is quantified by comparing the obtained estimation with an empirical one, which is the $(1 - \varepsilon)$th-quantile of the empirical distribution of the throughput of the complete trace. That is, this value represents the minimum capacity that should be allocated so that in only a predefined amount of time intervals of size T (i.e., ε) the throughput will be above the $C(T,\varepsilon)$. The empirical estimation is defined by:

$$C_{emp}(T,\varepsilon) := \min\left\{C : \#\{A_i \mid A_i > CT\}/n \le \varepsilon\right\} \quad,$$

where $A_1, \ldots, A_n$ are the empirical traffic aggregates at T. To verify whether the estimation was successful, we calculate the amount of measured intervals in which the traffic aggregate A_i exceeds $C(T,\varepsilon)$ by:

$$\hat{\varepsilon} := \#\{A_i \mid A_i > CT\}/n \quad.$$

where if $\hat{\varepsilon} \le \varepsilon$ the procedure yielded an acceptable estimation by not underestimating the required capacity.

Fig. 4 shows the average $\hat{\varepsilon}$ at $T = 10$ms and $T = 1$s for all traces per location and sampled at various rates. Again one can see that the shorter T and the higher the sampling rate, the lower $\hat{\varepsilon}$. That is, the error introduced by scaling the traffic rate and variance resulted in overestimation of required capacity for small timescales. This is also in line with results presented in Fig. 3. For example, for B traces, sampled 1:100 and at $T = 10$ms, the average estimation of required capacity mostly overestimates C_{emp}. For the same location and sampling rate, at $T = 1$s, more cases of underestimation were observed. Finally, when sampling rate is set to 1:1000, the overestimation becomes too large and for all locations $\hat{\varepsilon} = 0$ at $T = 10$ms.

V. CONCLUSIONS

In this paper we showed the feasibility of using sampled data using sFlow algorithm for link dimensioning. For a range of sampling rates the employed procedure successfully estimated the link required capacity. Clearly, to avoid the
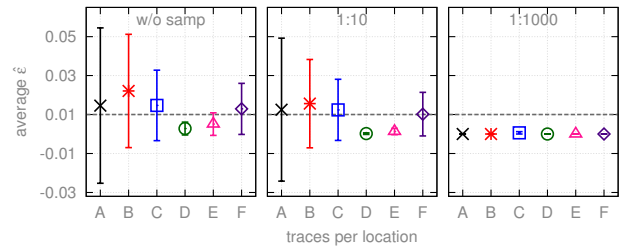
(a) $T = 10$ms
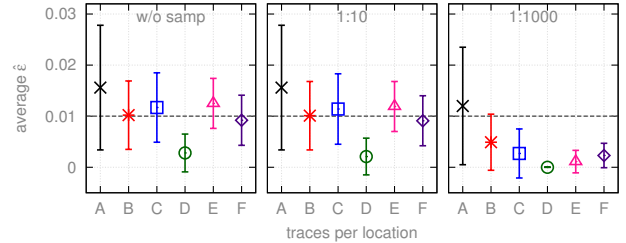
(b) $T = 1$s

Fig. 4. Average and standard deviation of $\hat{\varepsilon}$ for all traces per location at different sampling rates; $\varepsilon = 1\%$.

loss of important traffic properties such as Gaussianity and, consequently, inaccurate estimation of required capacity, an appropriate sampling rate should be chosen, i.e. consistent with the actual load of the monitored link. Our results also show that due to the random nature of the sampling algorithm, overestimation of traffic variance and required capacity is likely to happen at short timescales. However, overestimation within reasonable bounds is typically not crucial since it will not negatively impact on the QoS.

REFERENCES

[1] R. van de Meent, *Network Link Dimensioning: A Measurement & Modeling Based Approach*, PhD thesis, University of Twente, the Netherlands, 2006.

[2] A. Pras, L. J. M. Nieuwenhuis, R. van de Meent and M. R. H. Mandjes, *Dimensioning Network Links: A New Look at Equivalent Bandwidth*, IEEE Network, vol. 23, issue 2, pp. 5–10, 2009.

[3] P. Phaal, S. Panchen and N. McKee, *InMon Corporation's sFlow: A Method for Monitoring Traffic in Switched and Routed Networks*, RFC 3176, 2001.

[4] T. Zseby, M. Molina, N. Duffield, S. Niccolini and F. Raspall, *Sampling and Filtering Techniques for IP Packet Selection*, RFC 5475, 2009.

[5] B. Claise, G. Dhandapani, P. Aitken and S. Yates, *Export of Structured Data in IP Flow Information Export (IPFIX)*, RFC 6313, 2011.

[6] B.-Y. Choi, J. Park and Z.-L. Zhang, *Adaptive Random Sampling for Traffic Load Measurement*, in proc. of the 38th IEEE International Conference on Communications (ICC), vol. 3, pp. 1552–1556, 2003.

[7] B.-Y. Choi, J. Park and Z.-L. Zhang, *Adaptive Packet Sampling for Accurate and Scalable Flow Measurement*, in proc. of the 47th IEEE Global Telecommunications Conference (GLOBECOM), vol. 3, pp. 1448–1452, 2004.

[8] R. de O. Schmidt, R. Sadre and A. Pras, *Gaussian Traffic Revisited*, in proc. of the 12th IFIP Networking (to appear), 2013.

[9] B. M. Brown and T. P. Hettmansperger, *Normal Scores, Normal Plots and Tests for Normality*, Journal of the American Statistical Association, 91(436), pp. 1668–1675, 1996.