

Self-Management of Hybrid Networks: Can We Trust NetFlow Data?

Tiago Fioreze*, Lisandro Zambenedetti Granville[†], Aiko Pras*, Anna Sperotto*, Ramin Sadre*

*University of Twente, Design and Analysis of Communication Systems (DACS) – Enschede, The Netherlands

[†]Federal University of Rio Grande do Sul, Institute of Informatics – Porto Alegre, Brazil

Email: {t.fioreze, a.pras, a.sperotto, r.sadre}@utwente.nl*, granville@inf.ufrgs.br[†]

Abstract—Network measurement provides vital information on the health of managed networks. The collection of network information can be used for several reasons (*e.g.*, accounting or security) depending on the purpose the collected data will be used for. At the University of Twente (UT), an automatic decision process for hybrid networks that relies on collected network information has been investigated. This approach, called self-management of hybrid networks requires information retrieved from measuring processes in order to automatically decide on establishing/releasing lambda-connections for IP flows that are long in duration and big in volume (known as elephant flows). Nonetheless, the employed measurement technique can break the self-management decisions if the reported information does not accurately describe the actual behavior and characteristics of the observed flows. Within this context, this paper presents an investigation on the trustfulness of measurements performed using the popular NetFlow monitoring solution when elephant flows are especially observed. We primarily focus on the use of NetFlow with sampling in order to collect network information and investigate how reliable such information is for the self-management processes. This is important because the self-management approach decides which flows should be offloaded to the optical level based on the current state of the network and its running flows. We observe three specific flow metrics: octets, packets, and flow duration. Our analysis shows that NetFlow provides reliable information regarding octets and packets. On the other hand, the flow duration reported when sampling is employed tends to be shorter than the actual duration.

I. INTRODUCTION

With the increasing bandwidth demand by applications such as high-definition television (HDTV) [1], grid computing [2] [3], and large scale scientific experiments (*e.g.*, LOFAR project [4]), network providers are increasingly adopting hybrid network infrastructures that present higher capacity and more reliable data transmission. IP/optical hybrid networks are those that take data forwarding decisions simultaneously at both IP and optical levels [5]. This allows, for example, IP flows to be fully transported via lambda-connections at the optical level (lambda switching) bypassing the per hop routing decisions of the IP level. This directly improves the Quality of Service (QoS) offered by hybrid networks if compared to traditional IP networks. Big IP flows that overload the regular IP level, for example, may be moved to the optical level where they experience better QoS (*e.g.*, irrelevant jitter and larger bandwidth). At the same time, the IP level is offloaded and can better serve smaller flows.

There are currently two main approaches to manage lambda-

connections in optical networks [6]: conventional management and GMPLS signaling. In conventional management, a central manager (*e.g.*, a human administrator or an automated management process) directly accesses optical devices to create/release lambda-connections, as well as defines which IP flows should be moved to the optical level. In contrast, GMPLS signaling enables optical switches to exchange signaling messages to coordinate the creation of lambda-connections, thus avoiding the individual configuration of each device from an external manager. However, the decision on which IP flows should be moved to the optical level is still taken by human operators. The human decisions impact directly on the success of employing either centralized management or GMPLS signaling. For example, it may take hours (intra-domain) or even days (inter-domains) before a lambda-connection is established by human operators. In such long periods, several big IP flows could have been transported via lambda-connections, but due to the decision delay they remain being routed at the IP level.

At the University of Twente, a new approach for the management of hybrid networks has been investigated in order to overcome this drawback [7] [8]. Our approach, called self-management of hybrid networks, consists of obtaining information from the managed hybrid network in order to automatically select IP flows from the network level, as well as creating/releasing lambda-connections to transport such flows at the optical level. In this research, we are mainly concerned of identifying at the IP level the so called elephant flows, since they are responsible for most of the IP traffic despite being of small number if compared to other flows [9] [10].

In order to decide which flows to move to the optical level, our self-management approach observes three flow parameters: octets, packets, and duration. According to the values found for these parameters, the self-management process decides if an IP flow should be moved to/from the optical level. Therefore, the accurate retrieval of octets, packets, and duration, per flow, is critical for the quality of the self-management decision. If incorrect values are reported, lambda-connections could, for example, be allocated for non-eligible flows, or they could be released for flows that are fully active. Thus again, the accurate determination of octets, packets, and duration for each flow is crucial.

There are some solutions to collect flow information from managed networks. The Simple Network Management Protocol (SNMP) [11], for example, is extensively used to monitor

link utilization in border routers. Flow information via SNMP can be collected as well (*e.g.*, via RMON and RMON-II), but they are rarely employed in practice. NetFlow [12], on the other hand, is largely used to collect flow information from Cisco routers. Although initially developed by Cisco, today several network devices of other brands also employ the solution. In addition, NetFlow has been strongly influencing the definition of IPFIX [13], which is an IETF effort on standardizing flow-based measurements. In this research we focus our investigation on flow information collected using NetFlow, since it is the most popular employed solution for flow measurement.

In actual scenarios, information collected via NetFlow may suffer from non-expected factors and therefore can be inaccurately reported. Our initial analysis showed that there are three main factors that may influence the accuracy of the collected data: (i) misconfiguration (*e.g.*, wrong routing tables, wrong NetFlow parameters), (ii) differences in NetFlow implementations, and (iii) the usage of packet sampling in the NetFlow measurement process. From these three factors, we investigate in this paper how reliable is the NetFlow collected information depending on the sampling ratio employed. We focus on sampling because real high-scale networks employ sampling in their measurement processes in order to decrease the amount of processed data and therefore reduce the consumption of storage and processing power in the measuring solution.

The specific research question that motivates our investigation is: *Can NetFlow information on elephant flows, obtained from real networks, be considered reliable when sampling is used?* In order to answer that, we have collected and analyzed IP elephant flows from three different real networks that employ different sampling ratios: the University of Twente (UT) network, where no sampling is in fact employed; SURFnet (SN), the Dutch academic backbone, where a sampling ratio of 1 out of 100 packets is used; and finally on GÉANT (G), a European backbone that employs a sampling ratio of 1 out of 1000. From the collected flows, those that have been classified as elephant flows have been filtered and observed. As we are going to present, the different sampling ratios affect octets, packets, and duration in a different manner. Our study also generated some interesting side effects, such as the observations that there is a large amount of flows reported as zero in duration, as well as the fact that there is no direct link between flow duration and bandwidth consumption even for elephant flows.

The remainder of paper is structured as follows. In Section II we review the current state of the art on network measurement solutions. In Section III we describe our methodology by presenting the decisions and steps taken to make our analysis. In Section IV we present the effects of sampling on the observed metrics. Finally, we close this paper in Section V, where we draw our conclusions and future work.

II. RELATED WORK

Management of optical networks is an area of research that has been presenting interesting results. Since optical networks

usually provide bandwidth larger than conventional networks, with better QoS, traditional management approaches for QoS-enabled networks (like DiffServ and IntServ management) become less critical. Although presenting better QoS, optical networks and their management present additional challenges, such as handling the disruption of the communication paths. Rerouting is a technique used in selecting alternative paths when the default one is unavailable. Pro-active rerouting may establish backup links to cope with future disruptions. Nelakidit *et al.* [14], for example, proposes a fast local rerouting solution to quickly select alternative paths. Although important, most research in this area do not look IP and optical network in an integrated way: they usually tackle one layer at a time, which is restricting in the case of hybrid networks.

On another front, monitoring and measurement research has been quite active in the recent past. Thompson *et al.* [15] present a characterization of network flow-based usage and workloads on a commercial backbone. The authors analyze traffic data collected at one observation point on different time scales. Recently, Kim *et al.* [16] present a detailed analysis of flow-based traffic characteristics. The metrics that have been taken into account are packets, bytes, and port distribution. These researches, however, have not been carried out over actual networks, which we believe to be essential for the understanding of the effects of flow-based measurements.

In another work, Duffield *et al.* [17] investigate how to retrieve representative data coming from several locations where sampling is employed. They conclude that traffic can be represented multiple times in the collected traces, and the increasing use of sampling during measurement leads to some classes of traffic being poorly reported. More recently, Ribeiro *et al.* [18] used packet sampling as the measurement technique while mainly observing its effect on the flow size distribution. They also observed the effects on the packet counts, SYN information, and sequence number information. They concluded that TCP sequence number information is essential for accurate flow size estimation.

We believe that these works – both historical analysis of traces and comparison of diverse observation points – even if suitable for highly detailed studies, are missing one important dimension of analysis. In the majority, they present isolated studies, in which only one network is considered. Moreover, most of them focus their analysis on estimating precise flow size distribution or packet distribution. Few it is known about the effects of different sampling ratios on the flow metrics: octets, packets, and duration when considering real high-scale networks.

III. METHODOLOGY

The analysis of the effects of sampling over collected network information can be performed in different ways. Simulation tools, for example, could be employed to reproduce a network being measured using packet sampling. A controlled environment of a lab network (test bed) could be used too. However, none of these methods can 100% capture the real behavior of sampling on actual networks. In addition, many

other factors, such as background traffic, could influence the usage of sampling and they would most likely not be predicted when using simulation tools or controlled test beds. Moreover, we believe that background traffic can play an important role by significantly influencing the way packets are sampled, and therefore distorting the collected information. Considering that, we believe that analyzing the effects of sampling using data collected from real networks would provide more significant and relevant conclusions. In the remainder of this section we present the steps taken to collect traces, as well as how these traces have been processed in order to enable our analysis.

A. Exporting NetFlow records from network routers

NetFlow-enabled routers inspect¹ IP packet headers on transit in order to update an internal cache of so called NetFlow data. Such NetFlow data are then exported as NetFlow records from the local cache to remote computers, called flow collectors, whenever one of the following conditions occurs:

- A flow is inactive longer than an *inactive timeout*, *i.e.*, the flow record was not updated in cache due to no new packets received for that flow before the inactive timeout;
- A flow is active longer than an *active timeout*, *i.e.*, the flow record has been constantly updated in the local cache for a period of time longer than the active timeout;
- The FIN and RST TCP flags of an observed packet indicate that the flow finished;
- The local NetFlow cache is full and needs to be flushed.

By default, NetFlow routers are configured with an inactive timeout of 15 seconds and an active timeout of 30 minutes. It is important to notice that, given the existence of the active timeout, long lived flows (*e.g.*, elephant flows) will be reported by multiple (yet complementary) flow records. For example, if the active timeout is set to 30 minutes and a flow lasts 120 minutes, this flow will generate 4 NetFlow records, each one reporting a duration of approximately 30 minutes. In order to calculate the real duration of this flow, the 4 NetFlow records have to be combined. The step of flow combining is described in Subsection III-D ahead.

B. Packet sampling in flow-based measurements

The inspection of all packets seen by a router to update NetFlow data in the internal cache may be prohibitive [19] in such large networks due to limitations, for example, in the required router's CPU power. In order to overcome this issue, packet sampling is used instead. Instead of inspecting every packet in order to maintain NetFlow data, routers employing sampling inspect every n th packet, *i.e.*, there is a systematic sampling of 1 packet out of n (1: n). As a result of sampling, the final traffic volume, for instance, ends up being estimation rather than the actual measured flow volume. In the case of our specific experiments, the sampling ratios used in the

¹Packet inspection, employed in flow-based measurement, inspects network packets seen on the wire to update the appropriate fields of flow records. Inspected packets are not stored in the router's internal cache.

considered networks, as well as their inactive and active timeouts (seconds scale) are presented in Table I.

Network	Sampling ratio	Inactive timeout	Active timeout
UT	1:1	15	60
SURFnet	1:100	30	300
GÉANT	1:1000	60	300

TABLE I
PACKET SAMPLING RATIOS, AND INACTIVE AND ACTIVE TIMEOUTS PER CONSIDERED NETWORK.

C. Setup for collecting network traces

In order to collect NetFlow data from the considered networks, NetFlow-enabled routers in the UT [20], SURFnet [21], and GÉANT [22] networks have been configured to export NetFlow records to flow collectors hosted in the UT network. UT and GÉANT routers use NetFlow version 5 while SURFnet routers use NetFlow version 9. Despite this diversity, the collecting process has not been affected by the different NetFlow versions. Traces from the three networks have been collected over a period of one week (Jul 26th 2007 - Aug 3rd 2007). Figure 1 depicts the experimental setup used in our research.

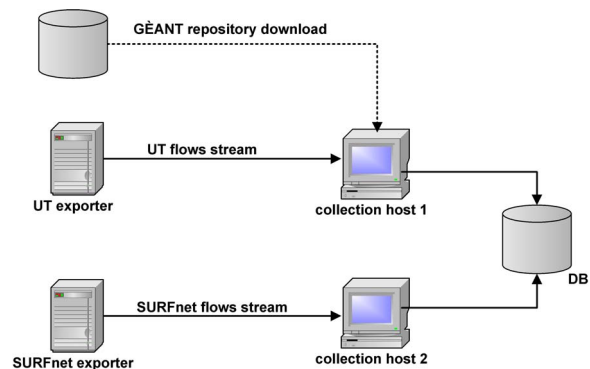


Fig. 1. Experimental environment setup.

The collectors have been set to dump the incoming NetFlow records in *pcap* files, using the *tcpdump* tool [23]. The decision to capture records in *pcap* format has been taken because it allows us to store them in their “raw state”, preserving all information forwarded by the exporter and in a format that does not limit our further analysis.

D. Combining NetFlow records

Once the traces are collected, they need to be refined in order to facilitate their analysis. To achieve that, we limited our analysis to 2 working days, which corresponds to approximately one-fourth of the total collected traffic. Our analysis covers therefore the period from Aug 1st 2007 00:00:00 UTC until Aug 2nd 2007 23:59:59 UTC.

In addition, since different definitions of flows are employed by the UT, SURFNet, and GÉANT NetFlow-enabled routers, we re-normalized the collected traces in order to use a common

set of flow fields. More specifically, we used source and destination IP addresses/TCP ports, and transport protocol to identify flows in the working traces. For a complete description of the fields existing in NetFlow version 9, see the work of Claise [24]. It is important to notice that other NetFlow routers, in other environments, could employ either broader or more restrictive flow definitions. A discussion about flow definitions and associated effects can be found in the work of Fioreze *et al.* [25].

Since NetFlow reports flow metrics in parts, one needs to combine the NetFlow records in order to closely compute the original flow duration, number of packets, and octets. In order to combine NetFlow records, there is a need to determine the gap that separates two consecutive flow records of the same flow. We have deliberately chosen a gap of 30 seconds, which is a common value for the TCP TIME-WAIT² state. We then decided that all NetFlow records of the same flow whose gap was smaller or equal to 30 seconds are grouped into the same flow. All our analysis were made then over combined flows rather than using NetFlow records.

E. Storing and filtering the collected data

Once the NetFlow records have been combined into flows, it is necessary to store the flows in a format suitable for our analysis. In our case, we imported the flows into a MySQL database [26]. MySQL has been chosen due to the familiarity of the paper's authors with such a tool. Our MySQL database consists of three different tables, one for each considered network. Each table contains several million of flows and requires a storage space in the order of tens of GB. Moreover, the analysis has been performed on MySQL 5.0 running in a 2xIntel Dual-Core Xeon 3.2GHz Linux Debian 4.0 machine, equipped with 4GB RAM.

Before starting our analysis, the data stored was filtered in order to select only flows that transited in the three considered networks and had therefore the chance of being sampled in these networks. Another filter was used over the first filtered data in order to select only elephant flows. We only consider elephant flows because they are the kind of flows that our self-management approach looks for. Moreover, elephant flows are small in amount, but they continuously generate most of the traffic in the observed networks. In our context, elephant flows are flows that, besides generating most of the traffic, have reported duration longer than 15 minutes. In our analysis, the elephant flows amounted to 7.42% of the total number of flows, but they generated 60% of the total observed octets. With this in mind, the coming section shows the results of our analysis considering elephant flows.

IV. TRACE ANALYSIS

This section presents the results of our analysis considering flow octets, packets, and duration. These metrics are consid-

²When a Transmission Control Protocol (TCP) connection is closed, the socket pair associated with the connection is placed into a state known as TIME-WAIT, which prevents other connections from using that source/destination IP addresses, source/destination TCP ports, and protocol for a period of time.

ered by comparing their expected value in theory with their obtained value in practice when sampling is employed. In addition, the NetFlow data collected from UT was considered as basis for this comparison. Table II shows how the expected values for each parameter are calculated for each network. Octets (O) and packets (P) observed in UT are expected to be reported in SURFnet and GÉANT as the original O and P divided by the sampling ratio employed, *i.e.*, 100 and 1000, respectively. In the case of duration (D), regardless the number of octets and packets seen in SURFnet and GÉANT, the same duration D is expected in both networks because the sampling ratio should not affect it.

Network	Octets	Packets	Duration
UT	O	P	D
SURFnet	O/100	P/100	D
GÉANT	O/1000	P/1000	D

TABLE II
EXPECTED VALUES IN THEORY FOR THE SURFNET AND GÉANT NETWORKS.

A. Flow octets

The first metric considered in our analysis is the flow octets. We compare this metric by its number expected in theory with the number obtained in practice (reality). Figure 2 shows the expected value in theory for SURFnet and GÉANT octets and the values that have been obtained in practice. The x-axis shows the elephant flows ordered by their number of octets in descending order, being therefore elephant flows with bigger sizes leftmost and smaller ones rightmost. The y-axis represents their number of octets in logarithmic scale.

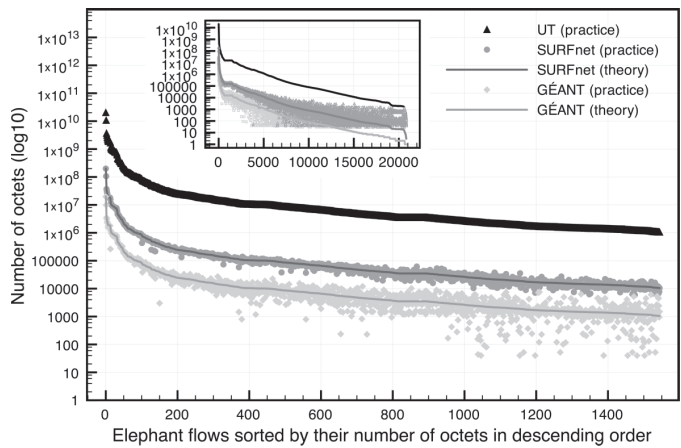


Fig. 2. Octet distribution in theory and in practice per organizational network.

Although the behavior of all flows can be difficult to predict, the behavior of elephant flows measured via flow sampling is closer to reality. The reason for that is that elephant flows generate a large amount of packets, which increases the chances of their packets to be sampled and, as a result, increases the chances of flow octets being accurately reported.

On the contrary, the remaining flows (mice flows) have a smaller number of packets, which make their packets less likely to be sampled, contributing therefore for their distortion of reality (see the small in-chart in Figure 2).

Equally important, Figure 2 shows that the bigger the elephant flows are, the closer their obtained values in practice are to their expected values in theory. On the other hand, when the size of elephant flow decreases, more imprecise are their reported values, becoming more similar to the behavior of the mice flows, but in a much smaller level of distortion when compared with them.

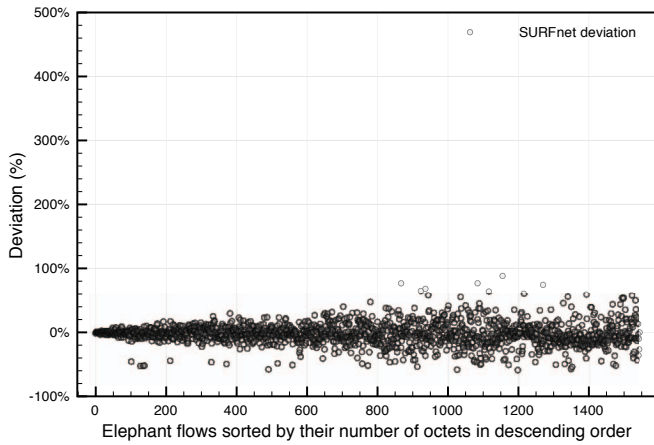


Fig. 3. Deviation percentage of SURFnet elephant flows when compared with its theoretical value.

Figures 3 and 4 show the deviation percentage of the SURFnet and GÉANT flows when compared with their expected value in theory. The average deviation found in our analysis was -0.55% in SURFnet and -5.47% in GÉANT.

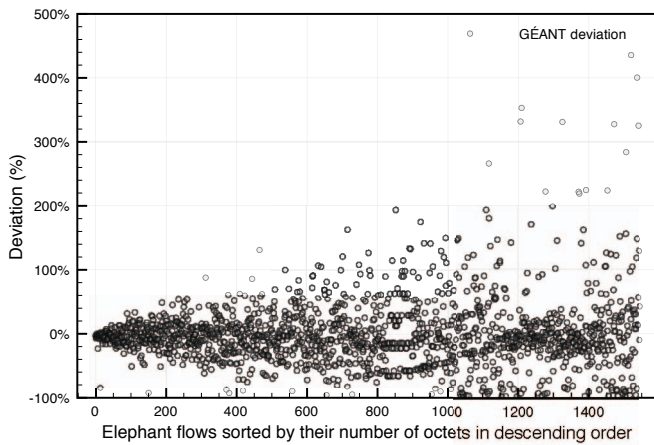


Fig. 4. Deviation percentage of GÉANT elephant flows when compared with its theoretical value.

B. Flow packets

As the second metric, we consider flow packets also comparing its number expected in theory with the number obtained

in practice. Figure 5 shows the expected value in theory for SURFnet and GÉANT packets and the values that were obtained in practice. The x-axis shows the flows ordered by the number of packets in descending order, being therefore the flows with the bigger amount of packets leftmost and the smaller ones rightmost. The y-axis is in logarithmic scale showing the number of packets. The figure shows that if instead of octets, we check the behavior in terms of packets, the behavior is rather similar.

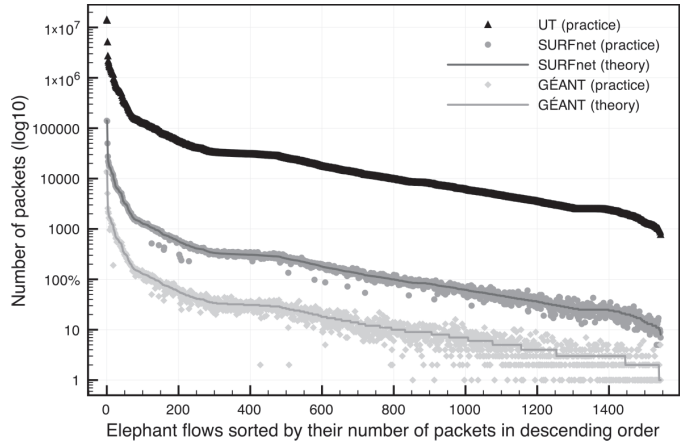


Fig. 5. Packet distribution in theory and in practice per organizational network.

Figure 5 shows that the bigger the amount of packets of elephant flows are, the closer the obtained values for the elephant flows packets are to their expected value in theory. This was also seen in the previous metric. The reason for that follows the same explanation presented before: the larger the amount of packets generated, the higher the chances are these packets are sampled.

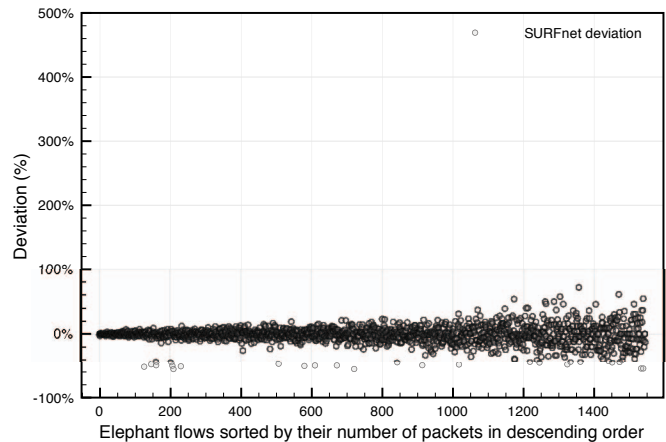


Fig. 6. Deviation percentage of SURFnet elephant flows when compared with its theoretical value.

Figures 6 and 7 show the deviation percentage of the SURFnet and GÉANT flows when comparing with their expected value in theory. The average deviation found in our

analysis is -0.62% in SURFnet, while in GÉANT it is -6.20%.

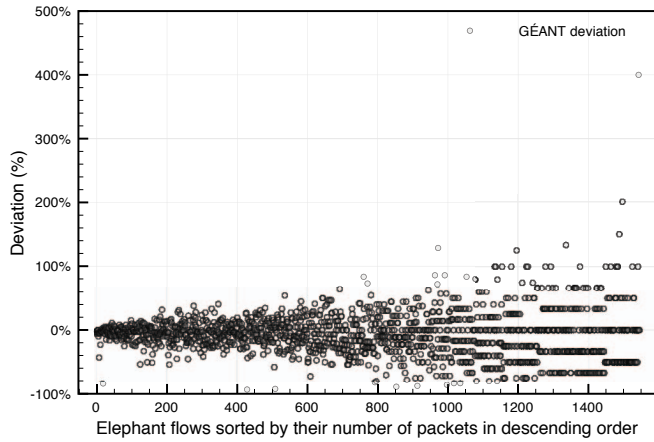


Fig. 7. Deviation percentage of GÉANT elephant flows when compared with its theoretical value.

Partial conclusion is that both parameters present a certain deviation from the reality, but their deviation is considered small in terms of percentage. SURFnet, which has a sampling ratio smaller than GÉANT is more precise, whereas GÉANT presents more deviation from the expected value.

C. Flow duration

This section presents the duration of the observed elephant flows from the three considered organizational networks. The flow duration is shown as total duration, which consists of the timestamp of the last seen packet belonging to the flow minus the timestamp of the first packet seen.

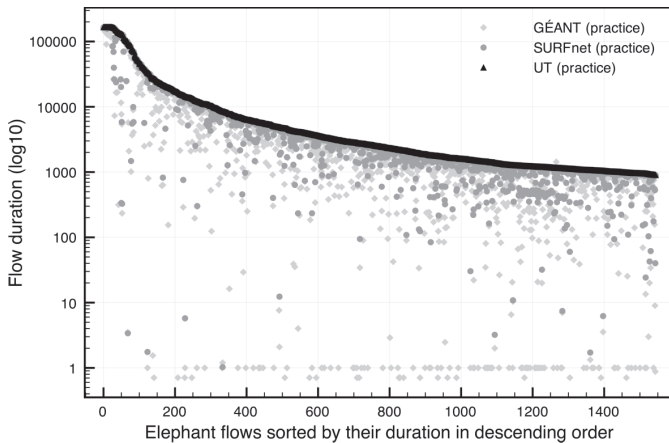


Fig. 8. Packet duration distribution per organizational network.

Figure 8 shows the obtained duration (in seconds scale) for SURFnet and GÉANT networks. In theory, they should have the same duration as UT, which does not happen. The x-axis shows the flows ordered by their duration in descending order, being therefore the flows with the longer duration leftmost and the shorter ones rightmost. The y-axis is in logarithmic scale showing the duration of every single elephant flow.

In order to better represent the data presented in Figure 9, we model it by using Power regression model. Power regression best fits our data to be as close to the actual data as possible. The best fit lines allow us to better see how much the flow duration obtained in reality deviates from the expected value in theory.

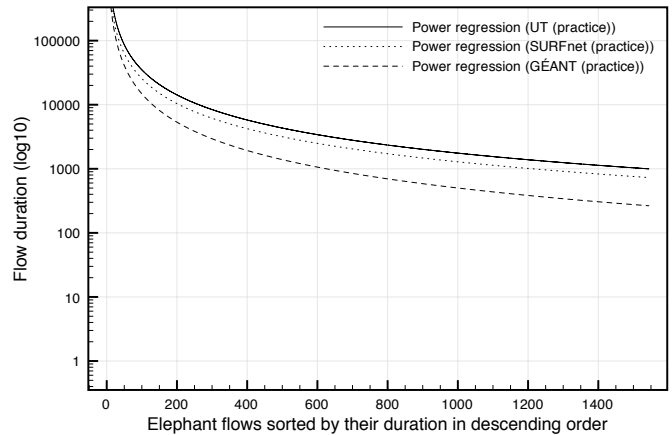


Fig. 9. Packet duration distribution per organization network using Power regression.

SURFnet reports, in average, -15.15% of difference to the UT duration, while in GÉANT the difference is in average -31.15%. This allows to conclude that the greater the sampling ratio is, the greater the number of misreported flows (in terms of duration) will be. The reason for that is that duration is more sensitive for missing packets than the packets and octets metrics. That happens when packets from the outermost of a flow are missed. For example, if a flow had 1000 packets whose the 900 first ones were regularly sent in intervals of 5 seconds and the last 100 packets were sent in intervals of 25 seconds. At its end, this flow would have in reality a duration of 7000 seconds. Let's imagine though that the 100 last packets of this flow were not sampled: the flow would then be reported as having 900 packets and a duration of 4500 seconds. That is, the number of packets would be 10% less than in reality, whereas the duration reported would wrongly be report as 35% shorter than in reality. Since packets are not equally spaced in time, this misinformation about the flow duration could even be more inaccurate. In addition to that, we saw, in our analysis, several flows whose reported duration was less than 50% of their real duration.

In order to understand the relationship, for elephant flows, between octets and duration, Figure 10 presents the same graph presented in Figure 8, but the x-axis is now ordered by octets. The figure allows us to observe that in fact there is no relationship between elephant flow's volume and duration. It means that some elephant flows generate lots of octets (big in volume) in a shorter period of time (small duration), while others may instead last longer but consume less bandwidth.

Partial conclusion regarding to the metric flow duration is that it is more risky for our self-management approach to

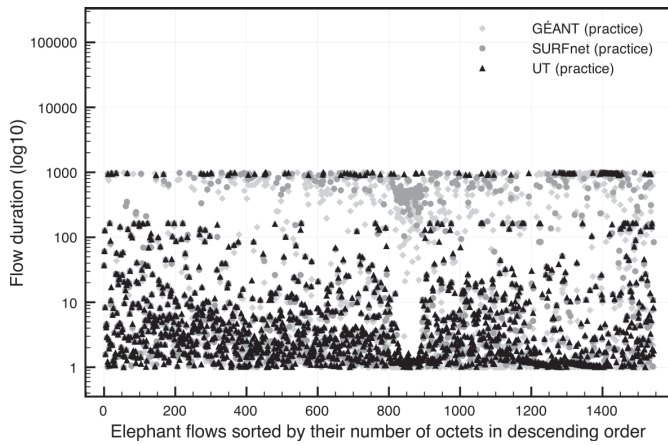


Fig. 10. Flows duration ordered by flows octets.

offload traffic based on reported flow duration than taking the same decision based on octets or packets.

We have observed the behavior of sampling considering flows that last at least 15 minutes. The reason is that flows that last less than 15 minutes are considered too short to be offloaded to the optical level by our self-management approach. This limit of 15 minutes in fact may be changed according to the policies on each hybrid network. In order to understand how flows that last longer or shorter than 15 minutes are reported in average, we varied the minimum duration from 1 to 180 minutes and computed the mean deviation for octets, packets, and duration for SURFnet and GÉANT networks. The result of that is presented in Figure 11.

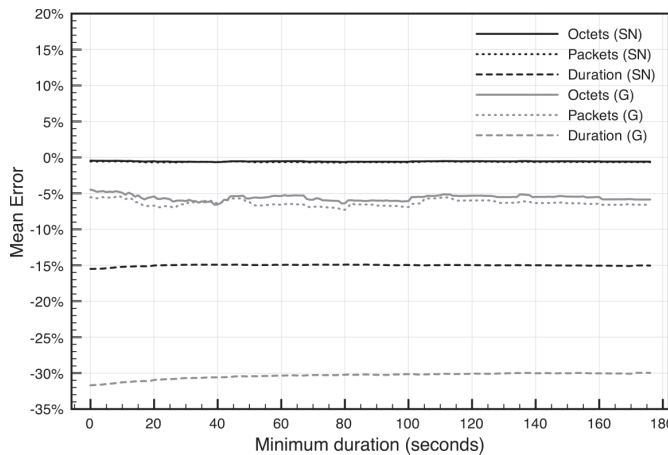


Fig. 11. Deviation percentage of SURFnet and GÉANT elephant flows while varying the flow duration.

As it can be seen, the higher sampling ratio of GÉANT affected the quality of the reported information. Again, the parameter that suffers the most is duration. It is also possible to observe that the deviation in SURFnet generally stabilizes earlier than in GÉANT due to the sampling ratios employed.

For example, in SURFnet flows that last at least 20 minutes are reported with a deviation similar to the set of flows lasting longer than that. In the case of GÉANT, however, the stabilization in terms of duration is reached at around 80 minutes.

Flows with duration equal to zero

Another interesting fact regarding to duration found during our analysis was the great amount of flows reported with duration equal to zero, reported in the three considered networks. The two main reasons for this large amount of flows with duration equal to zero are:

- *The existence of applications that regularly send control messages:* we have found that there is a group of applications that make NetFlow to generate flows with a single packet (*i.e.* flows with duration zero). We have seen applications such as the well-known DNS and NTP, but also some ones such as Gnutella. These applications periodically send control messages either for synchronization (*e.g.*, NTP) or for cache updates (*e.g.*, DNS). If this periodicity is bigger than the NetFlow inactive timeout (see Subsection III-A), this will result in thousands of flows with duration equal to zero.
- *The usage of sampling:* sampling increases the chance of not inspecting a packet belonging to an existent flow in cache. As a result of that, the number of flows with single packets is bigger in those networks where higher sampling ratios are used. For example, UT does not use sampling and it had 56.18% of the flows reported with duration equal to zero during the collection period. Within the same collection period, SURFnet (sampling ratio 1:100) and GÉANT (sampling ratio 1:1000) had 75.86% and 79.32% of the flows reported with duration equal to zero, respectively.

V. CONCLUSIONS AND FUTURE WORK

In this paper we presented a study about the impact on reporting network statistics about elephant flows when using NetFlow with packet sampling. This paper includes well-studied network statistics such as flow packets and octets, but also flow duration. Our analysis used three different organization networks as case study (UT, SURFnet, and GÉANT), as well as took into consideration their different sampling ratios (1:1, 1:100, and 1:1000, respectively).

The research question raised in this paper was *Can NetFlow information on elephant flows, obtained from real networks, be considered reliable when sampling is used?* Our conclusion is that the observed parameters octets and packets are reliably reported when sampling is used, but flow duration is considerably affected by its use. Our analysis show that, in average, octets and packets are respectively reported as -0.55% and -0.62% less than their expected value in theory (*i.e.*, they deviate) when a sampling factor of 1:100 (SURFnet sampling ratio) is used. Similarly, when a sampling factor of 1:1000 (GÉANT sampling ratio) is used, the average octets

and packets deviation from the expected value in theory is -5.47% and -6.20%, respectively. This leads us to conclude that the bigger the sampling ratio is, the less precise the number of octets and packets will be reported.

Not only octets and packets are affected, but also flow duration. Actually, flow duration showed in our analysis to be very sensitive to sampling. Several analyzed samples showed us a reported duration smaller than in reality. This imprecision in reporting the expected duration increases when the sampling ratio increases as well. Our analysis showed that in average the flow duration had a deviation of -15.15% and -31.15% from the expected value (*i.e.*, UT flow duration), when 1:100 and 1:1000 sampling ratios were used. The reason that flow duration is more sensitive to sampling than packets and octets comes from the fact that duration is highly affected when packets from the outermost of flows are not sampled. Even though this may not make a big difference to the overall amount of packets and octets belonging to a flow, it will pull down the duration of a flow, sometimes in seconds scales.

In addition, this sensitiveness increases when the number of packets belonging to a flow is small, that is, almost reaching the sampling ratio. Since the chances of a packet being sampled decreases in that case, it also increases then the chance that packets will not be tailed up to count for the duration (as well as for the packets and octets counters, but in a less harmful way). However, the opposite of that shows that the bigger the number of packets belonging to a flow is, more reliable is its duration, because the bigger are the chances the packet will be sampled and, therefore, counted.

Still regarding flow duration, the number of flows with duration equal to zero occurs due to the fact that some applications periodically send few packets either for cache updates or synchronization. This periodicity in sending packets makes NetFlow wait for the next packet of the flow, which does not happen. As a result, NetFlow exports this flow record with a single packet due to its inactivity. The usage of sampling also increased the number of flows reported as zero duration. The reason for that is because sampling increases the chances of a packet belonging to a flow in cache not to be selected and inspected. This flow record is as well exported due to inactivity, contributing then for the high amount of flows reported as zero duration. Finally, we also observed that in elephant flows, there is no relationship between flow volume and duration, since shorter flows may generate more traffic, while longer flows may consume less bandwidth.

As future work, it would be interesting to see more details about the influence of background traffic on the sampling data. This could help us to better understand how to overcome a situation where background traffic affects the sampled traffic of interest.

ACKNOWLEDGMENTS

This research work has been supported by the EC IST-EMANICS Network of Excellence (#26854). Special thanks to Roel Hoek (UT), Hans Trompert (SURFnet), and Maurizio Molina (GÉANT) for their valuable contribution in the flow

collection process. Also special thanks to the DACS students Daan van der Sanden, Gert Vliek, and Joris Kinable for their effort in the creation of our MySQL database. Last but not least, we thank Marcelo Edgar Böck for his help in the SQL queries.

REFERENCES

- [1] E. Baaren, L. van de Wijngaert, and E. Huizer, "Who's Afraid of High Def? Institutional Factors Influencing HDTV Diffusion in the Netherlands," in *ICDS*, 2008, pp. 42–48.
- [2] M. Toure, G. Berhe, P. Stolf, L. Broto, N. D. Palma, and D. Hagimont, "Autonomic Management for Grid Applications," in *PDP*, 2008, pp. 79–86.
- [3] R. Boutaba, W. Golab, Y. Iraqi, T. Li, and B. S. Arnaud, "Grid-Controlled Lightpaths for High Performance Grid Applications," *The Journal of Grid Computing*, vol. 1, no. 4, pp. 387–394, 2003.
- [4] K. V. der Schaaf, J. D. Bregman, and C. M. de Vos, "Hybrid Cluster Computing Hardware and Software in the LOFAR Radio Telescope," in *PDPTA*, 2003, pp. 695–701.
- [5] A. Leon-Garcia and I. Widjaja, *Communication Networks: Fundamental Concepts and Key Architectures*, 2nd ed. New York: McGraw-Hill Companies, 2003.
- [6] G. Bernstein, B. Rajagopalan, and D. Saha, *Optical Network Control: Architecture, Protocols, and Standards*. Reading: Addison Wesley Publishers, 2003.
- [7] T. Fioreze and A. Pras, "Using Self-Management for Establishing Light Paths in Optical Networks: An Overview," in *Poster session of EUNICE*, 2006, pp. 17 – 20.
- [8] T. Fioreze, R. van de Meent, and A. Pras, "An Architecture for the Self-management of Lambda-Connections in Hybrid Networks," in *EUNICE*, 2007, pp. 141 – 148.
- [9] T. Mori, M. Uchida, R. Kawahara, J. Pan, and S. Goto, "Identifying Elephant Flows Through Periodically Sampled Packets," in *ACM SIG-COMM*, 2004, pp. 115–120.
- [10] J. Wallerich, H. Dreger, A. Feldmann, B. Krishnamurthy, and W. Willinger, "A Methodology For Studying Persistency Aspects of Internet Flows," *Computer Communication Review*, vol. 35, no. 2, pp. 23–36, 2005.
- [11] D. Harrington, R. Presuhn, and B. Wijnen, "An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks," Request for Comments: 3411, Dec. 2002, IETF.
- [12] Cisco IOS NetFlow, <http://www.cisco.com/go/netflow>, May 2008.
- [13] B. Claise, "Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of IP Traffic Flow Information," Request for Comments: 5101, Jan. 2008, IETF.
- [14] S. Nelakuditi, S. Lee, Y. Yu, Z.-L. Zhang, and C.-N. Chuah, "Fast Local Rerouting for Handling Transient Link Failures," *IEEE/ACM Trans. Netw.*, vol. 15, no. 2, pp. 359–372, 2007.
- [15] K. Thompson, G. J. Miller, and R. Wilder, "Wide-Area Internet Traffic Patterns and Characteristics," *IEEE Network*, vol. 11, no. 6, pp. 10–23, 1997.
- [16] M.-S. Kim, Y. J. Won, and J. W. Hong, "Characteristic Analysis of Internet Traffic from the Perspective of Flows," *Computer Communications*, vol. 29, no. 10, pp. 1639–1652, 2006.
- [17] N. Duffield, C. Lund, and M. Thorup, "Optimal Combination of Sampled Network Measurements," in *IMC*, 2005, pp. 91–104.
- [18] B. Ribeiro, D. Towsley, T. Ye, and J. C. Bolot, "Fisher Information of Sampled Packets: An Application to Flow Size Estimation," in *IMC*, 2006, pp. 15–26.
- [19] N. Duffield, "Sampling for Passive Internet Measurement: A Review," *Statistical Science*, vol. 19, no. 3, pp. 472–498, 2004.
- [20] University of Twente, www.utwente.nl, May 2008.
- [21] SURFnet, www.surfnet.nl, May 2008.
- [22] GÉANT, www.geant.net, May 2008.
- [23] *Tcpdump/libpcap*, <http://www.tcpdump.org/>, May 2008.
- [24] B. Claise, "Cisco Systems NetFlow Services Export Version 9," Request for Comments: 3954, Oct. 2004, IETF.
- [25] T. Fioreze, M. O. Wolbers, R. van de Meent, and A. Pras, "Finding Elephant Flows for Optical Networks," in *IM*, 2007, pp. 627–640.
- [26] M. Siekkinen, E. W. Biersack, G. Urvoy-Keller, V. Goebel, and T. Plagemann, "InTraBase: Integrated Traffic Analysis Based on a Database Management System," in *E2EMON*, 2005, pp. 32–46.