On the efficient detection of elephant flows in aggregated network traffic

Javier Rivillo, José-Alberto Hernández and Iain W. Phillips
Networks and Control Group
Research School of Informatics
Loughborough University

Abstract

Studies have shown than best-effort delivery in Internet is not suitable for all traffic, where usually a few flows carry most of the data. This is commonly refered to as "elephant and mice phenomenon". The performance of the network can be improved detecting the elephant flows and applying traffic engineering solutions. However, current detection methods are not scalable or detect the elephant flows in a later than desirable.

In this paper we describe the importance and applications of elephant detection. Then we analyze real traffic data, collected by NLANR, to study the properties of elephant and mice flows. This analysis shows that the elephants are both heavy tailed and have a low mean packet interarrival time. These properties allow us to design a low computational cost, scalable detection method based on sampling windows. We present the method and some results obtained by its application.

1 Introduction

Since the early design of the Internet, network engineers have aimed to build a scalable and flexible network infrastructure, capable of growing and adapting to changes with little additional complexity. For this reason, algorithms and intelligent processing have traditionally been pushed out of the network core towards the edges, in an attempt to boost the performance of the inner network elements. As a consequence of such simplicity, networks have traditionally behaved in a best-effort policy, which implies no in-advance resource reservation nor differentiation in the treatment of packets.

However, today's Internet carries traffic from a wide range of applications, each of them with different requirements and constraints on network resources. Applications with special constraints such as video-conferencing, Internet telephony, on-line gaming, multimedia streaming and many more, have gradually appeared and the suitability of such best-effort policy and the need for providing different qualities of service to each type of application are being proposed.

Accordingly, it is typical to find many different flows with disparate characteristics, competing for the resources of the network. It is possible that some of the flows gain abusive use of the resources while all the rest share a fair use of them. With the simplicity of the design of the Internet at present, there is no possible way to prevent this. The actual analysis of network traffic has actually revealed this situation with the so-called *mouse* and *elephant* dichotomy. Such situation, if not taken under consideration, might lead to harmful levels of performance degradation.

The two types of flows referred to are those which make an extensive use of network resources (high bandwith-long-lived), the elephant flows, and those which do not consume so many resources, the mice flows. Typically, elephant flows consist of low-priority applications, i.e. large data transfer transactions and peer to peer file sharing. On the contrary, mice flows tend to be sensitive to delay jitter and high loss rates, which are mainly experienced in on-line gaming, small-sized web requests, multimedia broadcasting and voice over IP [1].

The accurate identification and special treatment of elephant flows, (either by rerouting, throttling, priority management) is crucial to guarantee a better performance of the global network and a higher user satisfaction.

This work introduces the basic guidelines in early identification of elephant flows based on their observed statistics (packet interarrivals and flow duration), and shows the benefits of assigning them special treatment. Section 2 gives some definitions and background. Section 3 shows some observations on the properties of both elephant and mice traffic flows. Section 4 proposes a means to identify elephant flows using a two-stage methodology. Section 5 shows a numerical example. Finally, sections 6 and 7 comprise the conclusions and further work.

2 Definitions and background

In network terms, a traffic flow consists of a unidirectional set of packets of the same transport protocol (either UDP or TCP) sharing the same source and destination IP addresses and ports. Accordingly, the totality of network traffic can be viewed as a superposition of multiple flows, each carrying different application traffic, from one side of the Internet to the other.

Qualitatively, elephant flows are streams of packets which contribute to network load substantially more than the rest of the flows. Typically, network managers and administrators define a threshold value to discern between elephants and mice. Obviously, such threshold value depends on the network-size. For instance, in a typical core router with more than 50000 flows traversing it per second, a flow that occupies, say 0.1% of the total traffic volume, can be considered as an elephant. However, in a local area network router with 1000 flows per second, such threshold may be too low.

Accordingly, three types of elephant flows are possible: elephants can be intensive and short-duration traffic streams (type 1), long-duration flows at low packet rates (type 2), or long-duration high-bandwidth (type 3), which are the biggest elephants. This is shown in figure 1, which depicts a typical cut of aggregated traffic over time, in a network router. Additionally, the same figure also shows examples of typical mouse flows.

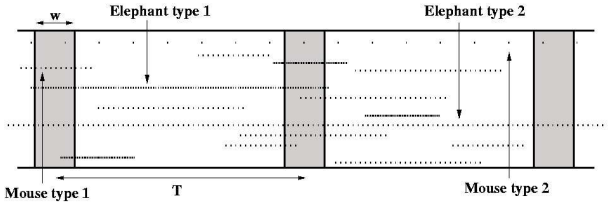


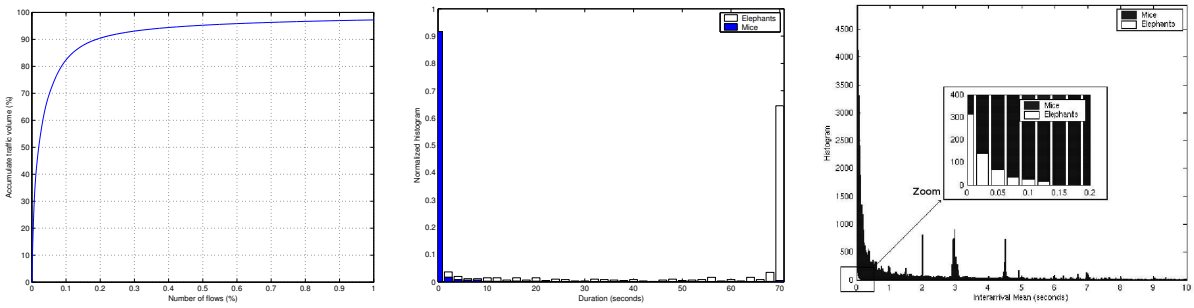
Figure 1: A flow aggregation view of network traffic

We can easily detect elephant flows sampling over time. In the example in Figure 1 the two elephant flows are the only flows which exceed two packets in both of the first two sampling windows. This is the basis of our detection method. In next section real traffic data is analysed to verify if this method can be extended to real traffic.

3 Analysis of traffic flows

We have considered a 70-second traffic trace collected by NLANR [2]. This is a backbone traffic trace is collected at the output of the Indianapolis router towards Cleveland. Further details on the hardware used and the measuring technique can be found at [3]. Such trace contains a total of 395,000 different flows carrying around 3.74Gbytes.

Figure 2 shows some statistics on this elephant traffic. Graph A shows the total contribution of the largest traffic flows The x-axis represents percentage of flows, and the y-axis shows the percentage of traffic volume. As shown only 0.1% of the traffic flows, that is 395 flows, contributes to nearly 83% of the total traffic. The next 0.9% biggest traffic flows (around 2555) contributes to an extra 13% of the traffic. The striking feature is that the remaining 99% flows (391050) carry only 4% of the total traffic.



A. Total contribution B. Flow duration histogram. C. Mean interarrival time.

Figure 2: The contribution of the 1% biggest flows in the total accumulated traffic

In the following, we will consider as elephant flows every stream which carries traffic exceeding 0.0182% of the total traffic, that is, 68Mbytes per flow in the 70-second trace. In average, such elephants flows consume around 1Mbyte/sec of the total bandwidth. This definition brings a total of 600 elephant flows in the trace.

Graph B shows a histogram of the typical duration of the flows. Black bars represent mouse flows and white bars stand for elephant flows. As observed, most of elephants are long-duration flows (type 2), since mouse flows rarely exceeds 5 seconds. Hence, if we monitor traffic at, say, every 15 seconds, it is very likely we will identify most of the elephant flows, with some (longer) mouse flows. However, there are a few elephant flows which are of less than 10 seconds duration. These ones are type 1 elephants, and shall be detected using other techniques.

For those type-1 elephant flows that are difficult to detect using flow duration, we can exploit another interesting feature of flows: packet interarrival. Graph C shows a histogram of the average packet interarrival time. As expected, elephant flows have short average packet interarrival time, typical from intensive applications. This feature of flows can also be exploited in designing an accurate detection strategy.

The next section introduces a low-computational two-phase method that exploits the properties of flow duration and average packet interarrival time to discriminate elephant flows from mouse flows.

4 A method to identify elephant flows

The methods published to identify elephants flows (see [4] and [5]) have either scalability problems because they require a very high computational cost which is increased with the speed of the network or the problem that they detect elephants at a late stage that doesn't allow traffic engineering solutions.

We propose a low computational cost methodology to detect the elephants using the properties of long heavy tail behaviour and short interarrival time with relatively low variance that we have discussed in the last section.

This process has two steps; the first is to monitor the flows in different time windows; and the second one is to process the information registered in the different sampling windows in order to decide which flows should be identified as elephants.

We are sampling the packet flows at a rate $Sr = T/w$ (see Figure 1). The sampling is suitable because the elephants have long tail behaviour, a large number of packets and a high rate of packet transmission. We propose that the distance between sampling windows is constant, but it may introduce correlation problems to periodic behaviour of specific flows with periodicity T seconds or multiples of T. This problem can be solved using a random distance between windows but with mean distance T.

The objective of the first phase is to analyze the flows in the current window in order to identify the suspected elephant flows and discard the rest as mice. In this way, we reduce the amount of memory and the processing being required for the next phase. The method we propose for this phase is to identify flows which reach or exceed a threshold of Np packets in the current window as suspected elephants. The reason to apply this method is that the elephants have the property of having a short mean interarrival time with low variance, so provided a suitable threshold (Np) and width of the window (w), the probability of identifying an active elephant flow is high.

In the second phase, the objective is to decide which of the suspected elephant flows of the current window are to be classified as elephants. The algorithm we propose is that a flow is identified as a elephant if the flow appears as a suspected elephant in a specified number (Nw) of different sampling windows. By tuning the parameters selection, the flow being identified as elephant has to have had a high rate of packets during Nw windows spaced out in the time, this implies it has a long tail behaviour and matches the properties of elephants.

5 Experiments and Results

The results of this detection method applied to the NLANR data are shown in the figure 3. In this experiment we have used sampling windows of 20ms and a sampling rate of 1%. With this sample rate we are able to detect most of the elephants in a few seconds. Once the are detected traffic engineering solutions may be applied to this flows identified as elephants.

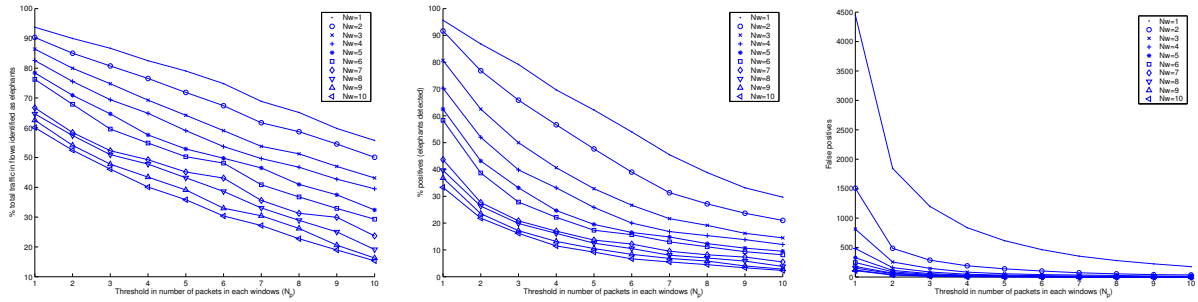
Figure 3 (Graph A) shows the results of traffic detected in the flows identified as elephants, positive ratio and false positive ratio for a wide combination of values for Np and Nw. If we choose $Np = 2$ and $Nw = 2$ the results are that 87% of the total traffic is carried in the flows identified as elephants, 78% of the elephants have been correctly identified and 0.15% of the mice flows are misidentified as elephant flows. Increasing Np and Nw we get more Precision but less Recall.

6 Further work

The next step of our work is proving the utility of detecting elephant flows and evaluating our method in this context. To this end we will simulate a network with a border router which has the capability to detect elephants and upon detection reroute the flows by an alternative path. In this way the delay would be reduced in flows belonging to applications with high priority, typically mice.

This study will be continued analyzing more real traffic flows. We are going to obtain our own traces from UK-Light [6] through MASTS [7]. This analysis can provide us a solid base to improve our detection algorithm.

We are considering some modifications in the detection method. The first one is include packet size information in the algorithm, which will reduce the number of false positives because the mice flows which are long lived with quite high packet transmission rate but smaller packet sizes. The number of false positives can be reduced further monitoring, even outside the sampling windows, the flows identified as elephants, to verify is they were really elephants. A further modification is to make this algorithm adaptive, in this way the algorithm's parameters will be adjusted automatically



A. Elephant traffic. B. Positive ratio. C. False positive ratio.

Figure 3: Flows identified as elephant traffic

according to the current traffic conditions and operator objectives.

It is a major objective to keep the algorithm scalable with low computational cost and memory requirements, so we must be careful including modifications which may compromise this goal. It is also desirable that the algorithm uses only a fixed amount of memory [8].

Finally, the users may adapt the shape of the elephant flows to avoid being detected as elephants. If the system is adaptive and the sampling windows are randomly placed the system is better protected from such users.

7 Conclusions

The utility of identifying elephant flows for traffic engineer solutions has been widely proposed. In a first stage we have studied real traffic data from NLANR [2] to obtain the properties of elephant and mice flows. The long tail behaviour and high packet transmission rate shown by the elephants have been used in the elephant detection method explained. This scalable and low computational cost method uses high sampling rate to all early detection of elephant flows. We have shown in the results that it is a valid method and its parameters may be adjusted for a tradeoff between Precision and Recall in identifying the elephant flows.

Acknowledgements

This work is undertaken under the UKLight MASTS project [7], which has brought together efforts from Loughborough University, Cambridge University and the University College, London.

The authors would like to acknowledge the provision of data from NLANR [2].

References

[1] N. Brownlee and K. C. Claffy, "Understanding Internet traffic streams: Dragonflies and tortoises," *IEEE Communications Magazine*, vol. 40, no. 10, October 2002.

[2] "The national laboratory for applied network research (nlanr)," `http://www.nlanr.net/`.

[3] "Abilene-i data set," `http://pma.nlanr.net/Traces/long/ipls1.html`.

[4] K. Papagiannaki, N. Taft, S. Bhattacharyya, P. Thiran, K. Salamatian, and C. Diot, "A pragmatic definition of elephants in internet backbone traffic," in *Proceedings of The Second Internet Measument Workshop IMW 2002*, ACM, Ed., 2002, pp. 175–176.

[5] Tatsuya Mori, Masato Uchida, Ryoichi Kawahara, Jianping Pan, and Shigeki Goto, "Identifying elephant flow through periodically sampled packets," in *Proceedings of The 2004 ACM SIGCOMM Internet Measurement Conference IMC 2004*, ACM, Ed., 2004, pp. 115–120.

[6] "Uklight," `http://www.uklight.ac.uk/`.

[7] "Masts (measurement at all scales in time and space)," `http://grid.ucl.ac.uk/MASTS.html`.

[8] Ken Keys, David Moore, and Cristian Estan, "A robust system for accurate real-time summaries of internet traffic," June 2005.