Measurement Tools and Techniques for Traffic and QoS Management

Philippe OWEZARSKI and Nicolas LARRIEU
LAAS-CNRS
7, avenue du Colonel ROCHE
31077 TOULOUSE Cedex 4
FRANCE
E-mail: {owe, nlarrieu}@laas.fr

Abstract - The Internet is on the way of becoming the universal communication network, and then needs to provide various services and QoS for all kinds of applications. But the large number of new applications generating new kinds of traffics, the fast increase of the number of users connected to the Internet, all new communications architectures and protocols, the heterogeneity in terms of technologies with fiber, wireless, satellite links, etc. make Internet traffic very complex, and very far from the common beliefs. The evolution of the Internet - that will have to guaranty multiple services - is then closely related to a better understanding of the traffic and of the global network behavior. Consequently, the development of measurement and monitoring based tools is nowadays the corner-stone for network engineering and research. This paper then aims at presenting measurement techniques and tools and illustrating how such measurements can be used for proposing new network protocols and architecture more suitable to current networking requirements. Especially, this paper will illustrate on a case study related to congestion control a "measurement based network engineering" method.
Keywords. Internet monitoring, traffic characterization, traffic management, measurement based network engineering

I. INTRODUCTION

The Internet is on the way of becoming the universal communication network for all kinds of information, from the simple transfer of binary computer data to the transmission of voice, video, or interactive information in real time. It has then to integrate new services suited to new applications. In addition, the Internet is rapidly growing, in size (number of computers connected, number of users, etc.), and in complexity, in particular because of the need of new advanced services, and the necessity to optimize the use of communication resources to improve the QoS[1] provided to users. In fact, the Internet has to evolve from a single best effort service to a multi-services network.

But there are many difficulties with the complexity of the Internet and all its network interconnections, with their resource heterogeneity in terms of technologies but also in terms of provisioning, and of course with the traffic characteristics. Indeed, because of the growing complexity of the Internet, all new applications with various and changing requirements, introduce in Internet traffic many characteristics that are very far from common beliefs. In fact, models with simple static metrics such as throughput, delay, or loss rate are really not sufficient to model completely and precisely Internet traffic dynamics that are its essential features. The evolution of the Internet is then strongly related to a good knowledge and understanding of traffic characteristics that will indicate the kind of mechanisms to deploy. Consequently, the development of monitoring-based tools and technologies to collect Internet traffic information, and methodologies to analyze their characteristics is currently an important topic for network engineering and research in networking.

This paper then aims at presenting measurement techniques and tools and illustrating how such measurements can be used for proposing new network protocols and architecture more suitable to current networking requirements. Especially, this paper will illustrate on a case study related to congestion control a "measurement based network engineering" (MBNE) method.

Thus, this paper is constructed as follows: first, section 2 deals with the description of the different existing measurement methodologies - respectively active and passive measurements techniques - as well as some related tools for illustrating them. Then, based on the measurement and traffic traces produced in the METROPOLIS[2] project, section 3 shows the characterization and analysis results of Renater traffic. In particular, this section puts forward that the new P2P applications, used most of the time to exchange large files

[1] QoS: Quality of Service.

[2] METROPOLIS is a French national project granted and funded by the French Network for Research in Telecommunications whose main goal deals with issuing new network monitoring and analysis methodologies. The network under consideration is RENATER, the French network for education and research that interconnects all universities, public research labs, some schools, as well as some industrial partners (depending on the project in relation with academia they are involved in).

as music or movies, are changing the characteristics of Internet traffic. New Internet traffic is now characterized by long range oscillations creating Long Range Dependence (LRD) in the traffic. LRD and oscillations are very damageable for the network QoS as they can provoke congestion, and provide very unstable services to users. Based on the results of traffic characteristics analysis, section 4 illustrates, on an example, how measurement results can be used for proposing new mechanisms for the Internet. In particular, as TCP congestion control mechanisms (that are not suited for the transmission of large files on high speed networks) are responsible of traffic oscillations and LRD, section 4 proposes to replace TCP congestion control mechanisms by TFRC[3], a congestion control mechanisms devoted to streaming applications and whose goal is to provide a smooth sending rate. The good impact of TFRC on traffic and network performance is then demonstrated, thus illustrating the interest of measurement based methodologies for network engineering. Finally, section 5 concludes this paper by presenting some future work for extending the "measurement based" methodologies.

II. Measurement techniques and tools

Measurements are then more and more considered as the cornerstone for network design, operation and research. But doing measurements is not an easy task. Indeed, depending on the point of view and requirements, different techniques and tools can be used. This section then introduces the two approaches existing for doing measurements that are active vs. passive methods, as well as the main tools that exist.

A. Active measurements

Active measurements consist in generating probe traffic in the network, and then observing the impact of network components and protocols on traffic: loss rate, delays, RTT, etc. This first approach for performing measurements is clearly oriented for providing measurements with a user point of view. Active measurements are the only way for users to measure the service parameters that they can take advantage of. On the other hand, the major drawback of this approach is related to the disturbance introduced by the probe traffic that can make the network QoS change, and thus provide erroneous measures. Much work addresses this issue of limiting the intrusiveness of probe traffic, trying to minimize its impacts on the network QoS as much as possible. This is, for example, one of the issues addresses in the IPPM[4] group of the IETF[5] [3, 4, 5, 22].

Active measurements are nevertheless very frequent in the Internet for which many tools for testing, validation and / or measurements are available. For example, it is possible to quote the very famous *ping* and *traceroute*. *Ping* aims at checking that a valid and alive path exists between two workstations, and measuring some parameters as RTT[6] or loss rate. *Traceroute* aims at showing the full set of routers crossed by probe packets from their source to their destination, and gives an indication on delays between every routers. *Ping* and *traceroute* are essentially used for measuring delays (even if *traceroute* is also the basic tool used for discovering the network topology). But making good end-to-end delays measurements implies to have synchronized clocks on all the probe machines. Such synchronization is very hard to enforce (NTP[7], for instance, does not provide good enough results on the Internet). That is why probe machines are often equipped with a GPS[8] card and antenna, as the reference time from the reference atomic clocks is broadcasted in the GPS system. And it seems that nowadays GPS is the only way to get an accurate enough synchronization of all probe machines for measuring delays. In addition, such system avoids all clock drift issues.

Another major issue addressed by active measurement techniques deals with measuring the available bandwidth on a path from a source and a destination workstation. Many tools have been designed for this purpose: cprobe [6], pathchirp [25], Spruce [26], PTR/IGI [12], pathload [13], Delphi [24], etc. To summarize, these tools are based on two main techniques: the first one consists in sending probe traffic burst to shortly fill the bottleneck link on the path and then get the available bandwidth on the path. The second technique consists in sending probe packets pairs and to measure the time difference between the two probe packets. Depending on the time difference between the two packets (and in particular how it has evolved), it is possible to compute the available bandwidth. Interested readers can refer to [26] for more details.

It also exists some active measurement environments as:

- NIMI[9] [23] that is an American infrastructure for active measurements. This infrastructure aims at being flexible and allows the collection of measurement results. All the active measurement tools quoted in this section can then be deployed on the NIMI platform, and some measurements can be performed using these tools between all nodes of the NIMI infrastructure. Note also that

[3] TFRC: TCP Friendly Rate Control.
[4] IPPM: IP Performance Metrics.
[5] IETF: Internet Engineering Task Force.

[6] RTT: Round Trip Time.
[7] NTP: Network Time Protocol.
[8] GPS: General Positioning System.
[9] NIMI: National Internet Measurement Infrastructure.

this platform has been extended towards Europe, in particular in Switzerland.

- RIPE that has deployed an infrastructure similar to NIMI in Europe. Compared to NIMI, RIPE provides some services to its customers: RIPE proposes to perform any measurement campaign requested by customers, in addition of the basic measurements that are continuously conduced (and whose statistical results are available for customer on a private web page. These basic results are related to delays and losses between all machines of the RIPE platform). Note however that the RIPE probe machine is sold as a set top box. It is not fully open, and it seems to be very hard to add new measurement tools.

B. Passive measurements

The principle of passive measurements consists in observing network traffic and to study its properties on one or more points of the network. As for active measurements, the two issues to address in the design of passive measurement probes are related to the transparency of the probe for the network and the traffic, as well as providing very accurate timestamps for traffic traces. By nature, passive measurements are not intrusive compared to active ones, as they do not generate any probe traffic. However, if the machine that hosts the passive probe has a limited capacity, and if the traffic crosses it, it can create a bottleneck in the network, and thus have a strong impact on network QoS, by delaying traffic and creating a congestion point. To avoid such impact of passive probe on network traffic, they are generally plugged to a splitter (thus letting traffic continue on its way, just receiving a copy of the ongoing traffic) or based on a bypass system that allows traffic transmission, even if the probe machine is overflowed or crashed.

One of the advantages of passive measurements is that they allow an endless range of analysis possibilities, including very advanced ones. But it is very difficult to determine the quality of the service that could be provided to users using passive measurements: passive measurements just provide a carrier point of view of traffic and network analysis.

Passive measurement systems can be also classified depending on the trace analysis mode. Thus, the system can perform on-line or off-line analysis. For on-line analysis, all the analysis related to one packet has to be complete before the arrival of the following packet. Such a real-time approach allows analysis on very long periods, and then provides significant statistics on the evolution of traffic. But the maximal complexity of the analysis remains limited because of the small amount of time to perform it.

On the other side, an off-line analysis forces the probe system to capture and store a traffic trace for a further analysis. Such an approach requires a huge amount of resources in terms of hard drives or storage systems, what represents a limitation for long duration traces. But, an off-line analysis allows very complex analysis, and then allows the analysis of very complex traffic properties. In addition, as traces are stored, it is possible to perform several different analyses on each trace, and to correlate all results for a better understanding of the complex network mechanisms.

Several very famous tools exist for doing passive measurements. The most famous tool is certainly *tcpdump* and all its relatives (*Etherreal*, *Windump*, etc.). *Tcpdump* is not really respecting the requirements for a passive measurement system as it is not transparent (if the dump machine is loaded, it introduces delay in the forwarding of each packet, and then the traffic is disturbed), and the accuracy of timestamps is very limited. But *tcpdump* is free, and then used a lot, and its storage format is generally taken by the other passive measurement tools.

But, in fact, the ideal place for doing passive measurements would be in routers. CISCO then developed *netflow* [7] in its routers, whose goal is to provide a trace of all flows passing in the router, as well as some statistics on these flows and the total amount of traffic. *Netflow* is very popular, and has been used in a lot of projects (even if it has been shown that *netflow* performances are not very good and that it impacts router performance). But it is widely deployed (present in all CISCO routers) and easy to use.

Similarly, some passive measurement tools are based on SNMP MIB. This is the case for instance of MRTG[10] [17] that is currently discussed at the IETF. MRTG then provides information on the traffic that went through a router, and this with the aim to be deployable on any routers.

Other equipments are not focusing on routers but on links. For instance, OC3MON and OC12MON are capture boards for IP/ATM OC3 and OC12 links respectively. The principle of these boards consists in capturing a trace of all packets propagating on a fiber link. The advantage of this card compared to what can be done with previous solution is that the capture is realized using hardware components at wire speed. The system is then completely transparent and, overall, able to capture complete traffic traces from high speed links. In addition, CAIDA[11] also developed a software tool suite called *CoralReef* [14] for analyzing such traces. As an example,

---

[10] MRTG: Multi Router Traffic Grapher.

[11] CAIDA: Cooperative Association for Internet Data Analysis.

note that OCxMON cards are used by Worldcom for monitoring the vBNS network in the United States.

With the same approach, ENDACE (a spin-off company of Waikato university in New Zealand), designed the same kind of boards but for many kinds of network technologies (Ethernet, Sonet, ATM, etc.) and not only ATM. These boards are called DAG. These cards are in charge of extracting for all packets passing on the link their header, timestamping it with a very accurate GPS clock and store it on a hard drive [8]. As for OcxMON, DAG is completely transparent and addresses links whose capacity can go up to OC192. In addition, the GPS timestamping is realized in the DAG card, using hardware, thus leading to a very high accuracy.

III. Traffic characterization

The DAG cards have been selected in the Metropolis project for capturing microscopic traffic traces on the Renater network. This section is then presenting the characterization and analysis results we got after analyzing the traffic traces that have been captured. To well understand the new traffic characteristics, it is first required to analyze the evolution of the Internet in terms of usages.

A. Evolution of traffic characteristics

The evolution of Internet traffic these last years has been marked by the huge increase of P2P traffic (Kaaza, e-donkey, …), and now, on some links of the RENATER Network, it can represent the same proportion than HTTP traffic (Figure 1). Such a result is quite impressive because, in an academic network as RENATER, students, teachers and researchers are not supposed to download music or movies. And, in fact, the amount of P2P traffic in RENATER is pretty low compared to the results observed on the commercial network of France Télécom[12], especially on the ADSL POP were P2P traffic can grow up to 70 % - and sometimes more!

Such an increase of P2P has necessarily an impact on traffic characteristics. In particular, because of the nature of file exchanged - mostly music and movies - that are very long compared to web traffic that was the dominant traffic in the Internet few years ago. In fact, the increase of P2P traffic, in addition of the classical traffic, makes the traffic have the following characteristics:

- There are always thousands of mice[13] in Internet traffic (because of the web, as well as P2P controls);
- But there are also a large number of elephants.

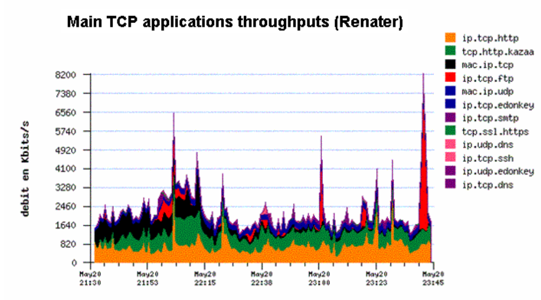

Fig. 1. Traffic distribution on the RENATER network in May 2003.

So, one of the main consequence of the evolution in terms of applications and usage is related to the flow size distribution changes. Figure 2 represents the flow size distribution between 2000 and nowadays. The exponential function (in black) is taken as a reference because the exponential distribution is closely related to the Poisson model that is most of the time used as the reference model for Internet traffic for simulations or performance evaluation. We can see on this picture that between year 2000 and nowadays the proportion of very long flows has increased in an important way. If in 2000, flow size distribution was almost exponential, this is completely wrong nowadays. Current distribution is very heavy tailed, and this distribution is then very far from the exponential distribution traditionally considered.

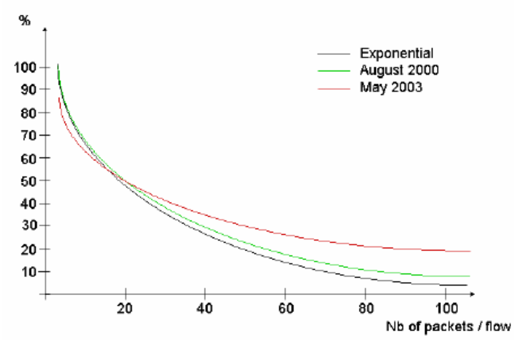

Fig. 2. Flow size distribution evolution between 2000 and 2003.

[12] France Télécom R&D is part of the METROPOLIS project, but the results got on the France Télécom network are not public and will not be discussed more in this paper.

[13] "Mouse" is a term used to designate a small flow, i.e. a flow that does not last enough to exit from the slow start phase of TCP.

B. Traffic Long Range Dependence and related issues

This increase of the proportion of P2P elephants hugely impacts traffic profile. Figure 3 illustrates it in current traffic. It shows the difference between the actual Internet traffic and Poisson traffic. These two traffics are observed with different granularities (0.01 s, 0.1 s and 1s), and it appears that Internet traffic does not smooth as fast as Poisson traffic when increasing observation granularity.

The analysis demonstrated that this result is completely due to elephants. In fact, the transmission of elephants creates in the traffic the arrival of a large wave of data that has the particularity of lasting for a long time - more than 1 second - while web flows are generally transmitted in less than one second on the current Internet. That is why we have this difference between Poisson and real traffic: the nature of oscillations between the two traffics changes, with oscillations in actual current traffic that are persistent in the network.

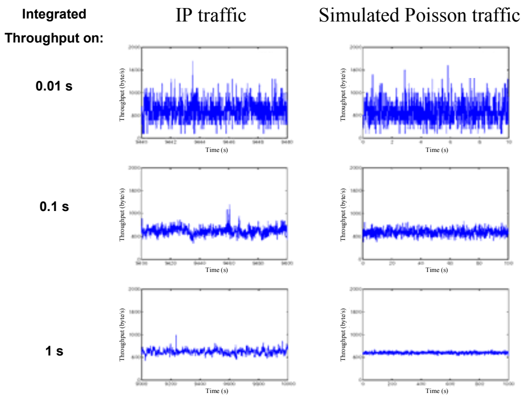

Fig. 3. Comparison between oscillations of Internet and Poisson traffics.

In addition, as TCP connections used for transmitting larger flows are longer, the dependence that exists between packets of the same connection propagates on longer ranges. This phenomenon is usually called Long Range Dependence (LRD) or long memory.

It has several causes, and in particular congestion control mechanisms deployed in the Internet, especially the ones of TCP, this protocol being the dominant protocol in the Internet [19]. Among all the TCP mechanisms, it is obvious that its closed control loop introduces dependence, as acknowledgements depend on the arrival of a packet, and the sending of all the following packets of the connection depends on this acknowledgement. In the same way, the two TCP mechanisms - slow-start and congestion avoidance - introduce some dependence between packets of different congestion control windows. By generalizing these observations, it is obvious that all packets of a TCP connection are dependent the ones from the other. In addition, with the increase of the Internet link capacities that allows the transmission of longer and longer flows, it is obvious that the range of the LRD phenomenon increases. That is why the persistence of the Internet traffic oscillations measured, even with a coarse granularity, is so high. Indeed, because of the TCP dependence phenomenon propagating in the traffic thanks to flows (connections), the increase of flow size also makes the dependence range increase and propagates on very long ranges. An oscillation at time t then provokes other oscillations at other time being potentially very far from t. A (short term) congestion due to a huge oscillation of a connection can then continue to have some repeats several hours later (in the case of a movie download for instance), i.e. this flow will continue to propose to the network some traffic peaks directly dependent from this first oscillation, and can create some new short term congestions. Moreover, it is clear that elephants, because of their long life in the network, and because of the large capacities of networks - most of the time over-provisioned - have time to reach high values of the congestion control window (CWND). Thus, a loss induces a huge decrease, followed by a huge increase of the throughput of the flow. The increase of flow size then favors high amplitude oscillations, dependent on very long ranges.

Of course, oscillations are very damaging for the global utilization of network resources as the capacity released by a flow that experiences a loss (for example) cannot be immediately used by another flow (because of slow start for instance): this corresponds to resource waste, and introduces a decrease of the global QoS of the network. In fact, the more the traffic oscillates, the lower the performances [20].

To give a concrete view of LRD issues on traffic, Figure 4 aims at illustrating it on a simple traffic parameter: losses. Figure 4.a depicts a leacky bucket as an analogy with a router for instance, its buffer, ingress and egress links. So, when there are waves in the arriving traffic (Figure 4.b), and if the goal is to provide a good service with no extra losses and no extra delays, it is first required to over-provision the link (otherwise the traffic will be smoothed, and at least, delays will be introduced for some packets). The second characteristic appears in the buffer when a wave arrives: it makes the level of the buffer increase (Figure 4.c). This is a well known issue of networking addressed many time before, especially by [15]. But when the range of oscillations increases (Figure 4.d), and this is the case with current Internet traffic, the arrival of a persistent wave provokes a buffer overflow, thus leading to losses (Figure 4.e).

Usually, in such case, network operators decide to increase buffer sizes. But this is not the right solution. Increasing buffer size makes delays and then RTT increase. Then TCP performance decreases and the transmission of the flow lasts longer, thus creating more LRD, and then more troubles. The approach that was most of the time used in the Internet and that consists in managing buffers of network components in order to be able to cope with traffic characteristics is then no more applicable. As flow size cannot be reduced (it is a characteristic of Internet users), the only solution is too modify protocols behaviors to decrease the range of waves generated by applications. This is one of the research directions that will be developed in section 4.

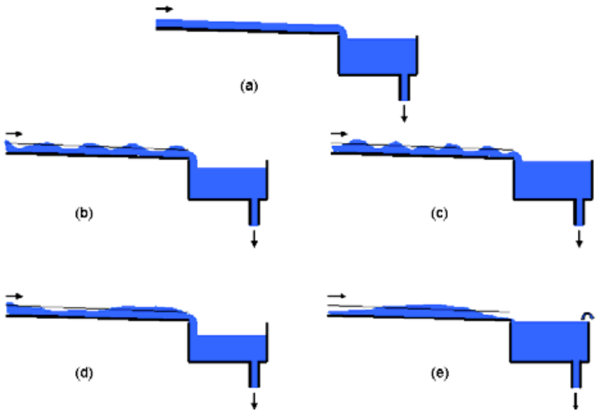

Fig. 4. Illustration of LRD issues on losses.

What has been shown in what precedes just presents some qualitative results on the nature of Internet traffic. But it is also needed to quantify its oscillating nature and its LRD characteristics. For that, we use a wavelet based analysis of the traffic [1]. The wavelet function has been selected as it is the best way to represent an oscillation. The principle deals with extracting from the traffic all wavelets. For that we use several wavelet functions with different frequencies to get the different range of variability of the analyzed traffic. The different wavelets represent different ranges of oscillations. The waves with the largest periods represent the very long waves, i.e. the ones generated by elephant flows. Interested readers can refer to [2].

The curve on Figure 5 has been obtained using the LD Estimate tool [27] that estimates the LRD that appears in Internet traffic at all scales. The output of this tool is a graphical representation of the dependence laws at all time scales. It represents the result of a wavelet based analysis of Internet traffic (and it is important to note that the same qualitative result as been exhibited for all links that have been monitored all over the world by researchers working on Internet links monitoring). It represents the variability of the oscillations depending on the observation range. Also note that the Hurst factor H that is the factor fully characterizing a self-similar process - and Internet traffic is often said to be self-similar [16, 21] - can be obtained directly depending on the slope of the LRD curve. The curve on figure 5 shows a bi-scaling phenomenon (in a log-log scale), with an elbow around octave 8, which shows a difference in the LRD level between short and long time scales for the traffic exchanged, and meaning that there are two different power laws. For short scales (octave < 8, left part of the curve), representing the dependence between close packets (i.e. packets whose sending time are not very far from each other), the dependence is quite limited. Such dependence is the one that can exist for packets belonging to the same congestion window and that are then very close from each other. On the other side, for long time scales (octave > 8, right part of the curve), LRD can be very high. For octaves 8 to 12 that correspond for instance to the dependence between packets of consecutive congestion windows, the dependence is higher. This can be explained by the closed loop of TCP congestion control mechanism in which the sending of one packet of a congestion control window depends on the receiving of the acknowledgement of one packet of the previous congestion control window. Of course, this phenomenon exists for consecutive congestion window, but also for all congestion windows of the same flow. This means that the presence in the traffic of very long flows introduces very long scale dependence phenomenon, as depicted on figure 5 for very large octaves. The consequence of such LRD is one major issue as every oscillation at time t will be repeated at any other time t' that is dependent from t (because of the long range dependence between packets due to protocols - here TCP - on long flows).

Note however that some additional experiments also show that the elbow in the curve corresponds to the mean size of flows, meaning that the right part of the curve corresponds to the impact of elephant flows.

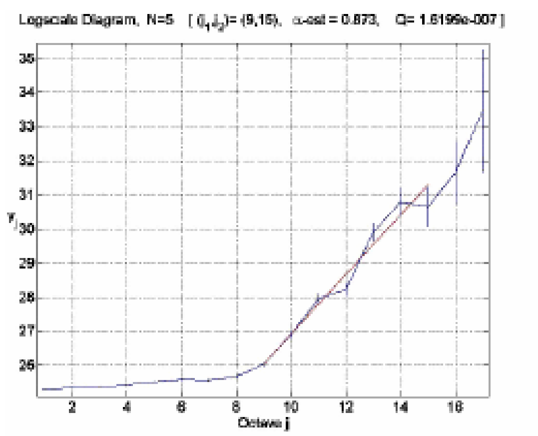

Fig. 5. LRD evaluation for edge network traffic.

IV. A PROPOSAL FOR IMPROVING CONGESTION CONTROL

Section 3 presented the analysis results of traffic traces, thus explaining its characteristics, and the evolution of these characteristics. This analysis then provides rich information on the behavior of TCP on elephants, and its failure. This section then illustrates our measurement based traffic engineering methodology. Indeed, based on the previous measurement analysis results, we are proposing a new mechanism for transmitting elephant flows in the Internet.

A. Increasing QoS by smoothing flow behaviors

It is clear now that network traffic has complex and high oscillating features. Indeed, it clearly appears the presence of scale laws in the traffic that induce the repetition of an oscillating phenomenon. From this observation, it appears that the most urgent problem to address deals with reducing oscillations and more precisely with regulating the long term oscillations having such a damaging effect on traffic QoS, stability, and performance. Therefore, the main objective is then to bring more stability to elephant flows.

To increase elephant flows regularity (i.e. to suppress observable oscillating behaviors at all scales), the new TFRC congestion control mechanism seems to be able to provide a great contribution.

B. TFRC principles

TFRC has been designed to provide a service suited for stream oriented applications and then aims at proposing to applications a smooth sending rate with very soft increases and decreases; at least much softer than the ones of TCP.

By associating such a congestion control mechanism to elephants, i.e. to the main part of the traffic, we expect to be able to control traffic oscillations, and then to increase global QoS and performance of the network. The sending rate of each TFRC source is made thanks to a receiver oriented computation, that calculates, once by RTT[14], the sending rate according to the loss event rate measured by the receiver [10, 11] according to equation 1:

$$X = \frac{s}{R \times \sqrt{2 \times b \times \frac{p}{3}} + \left(t_{RTO} \times \left(3 \times \sqrt{3 \times b \times \frac{p}{8}}\right) \times p \times \left(1 + 32 \times p^2\right)\right)} \quad (1)$$

where:

- X is the transmit rate in byte/second,
- s is the packet size in byte,
- R is the round trip time in second,
- p is the loss event rate (between 0 and 1.0), of the number of loss events as a fraction of the number of packets transmitted,
- t_{RTO} is the TCP retransmission timeout value in second,
- b is the number of packets acknowledged by a single TCP acknowledgement.

In TFRC, a loss event is considered if at least one loss appears in a RTT. This means that several losses appearing in the same RTT are considered as a single loss event. Doing so, the loss dependence model of the Internet is broken since most dependent losses are grouped in a same loss event. Thus, the recovery will be eased and more efficient compare to what TCP can do: it is well known that TCP is not very efficient to recover from several losses in sequence. This approach follows the results of [28] that proposes an analysis and a model for the Internet loss process.

C. Experiment description

Our experiment aims at providing a comparative evaluation of the global traffic characteristics if elephants use TCP or TFRC as the transmission protocol. This experiment aims at providing values in a realistic environment. For that, of course, the experiment relies on the use of traffic traces grabbed thanks to passive monitoring tools as the DAG [9] equipments. Therefore, traffic flows identified in the original traffic trace are replayed in NS-2 with the same relative starting date and the same others characteristics. Elephant flows are transmitted in the simulator using TFRC while other flows use TCP New Reno[15]. Then in the remainder, the comparative study will focus on the original trace and the simulated one where elephants are generated using TFRC.

In addition of the classical traffic throughput parameter, this study focuses on QoS statistical parameters as the LRD (as justified in section 3) and some parameters related to variability. For that, we used the Stability Coefficient (SC) defined as the following ratio:

$$\text{Stability Coefficient (SC)} = \frac{\text{exchanged average traffic}}{\text{exchanged traffic standard-deviation } (\sigma)} \quad (2)$$

D. TFRC impact on flow QoS

Figure 6 presents the traffic in both cases, i.e. in the real and simulated cases. It visually clearly appears that using TFRC for sending elephants, instead of TCP, makes

[14] RTT: Round Trip Time.

[15] TCP New Reno has been selected as it is currently the most used version of TCP in the Internet. To increase again the realism of simulations, it would be interesting to replay short flows with the same TCP version than the one that was used in the original trace, but finding out such information is impossible for most of short flows: only the ones that experiment a huge number of losses can provide enough information to find out the TCP version that was used.

global traffic much smoother, avoiding all the huge peaks that can be seen on the real traffic.

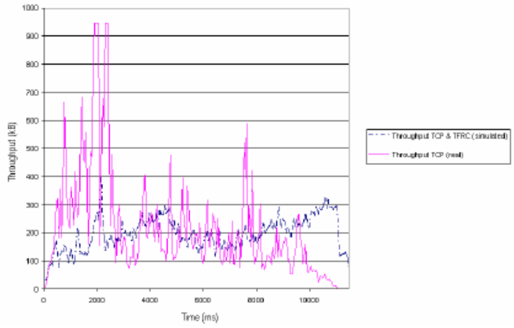

Fig. 6. Throughput evolution during time.

Quantitatively speaking, results are indicated in table 1. This confirms that the traffic variability in the case of real traffic (using TCP for transmitting elephants) is much more important compared to the simulated case in which elephants are transmitted using TFRC (for the standard deviation σ it has been calculated that σ(real traffic) = 157.959 ko >>> σ(simulated traffic) = 102.176 ko). In the same way the stability coefficient is less important in the real case (SC = 0.521) than in the simulated one (SC = 0.761).

Dealing with the global throughput we got for both real and simulated traffic rather equal values (Throughput(real traffic) = 82.335 ko ≈ Throughput(simulated traffic) = 77.707 ko). This result is quite good as TFRC is not able to consume as many resources as TCP [18], and even if TFRC is less aggressive than TCP, it is able to reach the same performance level as TCP. This confirms the importance of stability for good performances [20].

TABLE I

THROUGHPUT EVOLUTION DURING TIME FOR TCP AND TFRC PROTOCOLS

Protocol	Average throughput (kB)	Throughput σ (kB)	SC
TCP New Reno (NR): real case	82.335	157.959	0.521
TCP NR & TFRC: simulated case	77.707	102.176	0.761

Speaking about LRD in the simulated case, figure 7 shows that the bi-scaling property of the curve is strongly decreased, and that the curve has a very small slope. This means that all kinds of dependences, especially the long term ones have been drastically reduced. The values for the LRD (Hurst factor are: (H(real traffic) = 0.641 and H(Simulated traffic) = 0.194).

Such result confirms two aspects of our proposal:

- TFRC helps to smooth individual flow traffic (thus providing a smoother QoS better suited for stream oriented applications) as well as the global traffic of the link;
- LRD is the right parameter to qualify and quantify all scaling laws and dependencies between oscillations.

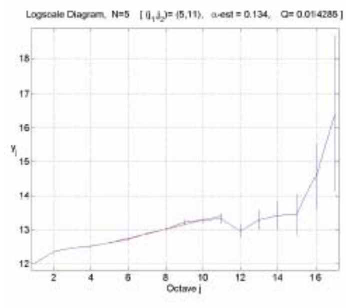

Fig. 7. LRD evaluation for simulated traffic including TFRC elephants.

V. CONCLUSION

As a conclusion, the main contribution of this paper was to present an overview of a new method for network engineering based on measurements. For this purpose, this paper started from the description of the measurement components that have to be designed and deployed in the network(s). It then presented some analysis of the traffic traces got by the measurements tools, especially describing and analyzing the oscillating nature of Internet traffic, and showing the impact of new usages related to P2P applications and the exchange of large files as music or movies. Finally, this paper showed through a case study on congestion control that traffic analysis results can help to propose new solutions that are much more suited to current network traffic and usages, thus illustrating the benefits of the "measurement based network engineering" method.

But some evolutions of the MBNE method are already planned. Indeed, the traffic characterization and analysis results show that traffic and its evolution is really unstable and unpredictable, and this is largely increased by applications and users behaviors. As a consequence, it is most of the time impossible to propose new mechanisms (as it has been proposed by replacing TCP by TFRC) that are valid in all cases. Most of the time, mechanisms have to adapt in real time to the current traffic load and characteristics because of the huge variability and diversity of traffics. As a consequence, the global measurement system has to be redesigned to be able to give traffic information in real time to all network components being in charge of the transmission, routing, security, charging, billing, etc. of the traffic. Some mechanisms for broadcasting (or at least multicasting) information to network components are required. Current researches mostly rely on some P2P topologies and architectures. And finally, depending on the measurements done in real time, it is needed to find the better reaction for the network components. In fact, the evolution that is in progress makes us move from a "measurement based network engineering" to a "measurement based networking" method, in which network components have to react in real time to the network state, from which they have information based on the measurements that are performed. Such a new method, of course, will require adapting the measurement system to the new needs, as well as the network components, as routers for instance. In particular, this approach has the advantage of giving to network components information not only from what happens in their close neighborhood, but also from other distant places. We expect that such information can help us to propose solutions, for instance for guaranteeing QoS, with a global point of view, and not only with a point of view limited to a single AS[16]. And in the framework of the Internet, on which we have very limited control, it seems to be the right direction to go.

REFERENCES

[1] Abry P. and Veitch D., ``*Wavelet Analysis of Long Range Dependent Traffic*'', Trans. Info. Theory, Vol.44, No.1 pp.2-15, Jan 1998.

[2] Abry P., Veitch V. and Flandrin P., ``*Long-Range Dependence: Revisiting Aggregation with Wavelets*'', Journal of Time Series Anal., Vol.19, No.3 pp.253-266 May 1998.

[3] G. Almes, S. Kalidindi, M. Zekauskas, *"A One-way Delay Metric for IPPM"*, RFC 2679, September 1999.

[4] G. Almes, S. Kalidindi, M. Zekauskas, *"A One-way Packet Loss Metric for IPPM"*, RFC 2680, September 1999.

[5] G. Almes, S. Kalidindi, M. Zekauskas, *"A Round-trip Delay Metric for IPPM"*, RFC 2681, September 1999.

[6] R. Carter, M. Crovella, *"Dynamic server selection using bandwidth probing in wide area networks"*, Technical Report TR-96-007, Boston University Computer Science Department, 1996.

[7] Cisco, *"NetFlow Services Solutions Guide"*, http://www.cisco.com/univercd/cc/td/doc/-cisintwk/intsolns/netflsol/

[8] J. Cleary, S. Donnelly, I. Graham, A. McGregor, M. Pearson, *"Design principles for accurate passive measurement"*, Passive and Active Measurements (PAM), Hamilton, New Zealand, April 2000.

[9] S. Donnelly, I. Graham, R. Wilhelm, ``*Passive calibration of an active measurement system*'', in Passive and active measurements workshop, Amsterdam, April 2001.

[10] Floyd S. and Fall K., ``*Promoting the use of end-to-end congestion control in the Internet*'', In Proc. IEEE ACM Transactions on Networking, 14 pages, February 1998.

[11] Floyd S., Handley Mr., Padhye J. and Widmer J., ``*Equation-based congestion control for unicast applications*'', In Proc. ACM SIGCOMM, 14 pages, 2000.

[12] N. Hu, P. Steenkiste, *"Evaluation and characterization of available bandwidth techniques"*, IEEE JSAC special issue on Internet and WWW measurements, mapping, and modeling, 2003.

[13]M. Jain, C. Dovrollis, *"Pathload: a measurement for end-to-end available bandwidth"*, Passive and Active Measurements (PAM'2002), Fort Collins, CO, USA, march 2002.

[14] K. Keys, D. Moore, R. Koga, E. Lagache, M. Tesch, K. Claffy, *"The Architecture of the CoralReef: An Internet Traffic Monitoring Software Suite"*, Passive and Active Measurements (PAM), Amsterdam, The Netherlands, April 2001.

[15] L. Kleinrock, *"Queing Systems Theory"*, Wiley, 1975.

[16] W. Leland, M. Taqqu, W. Willinger, D. Wilson, *"On the self-similar nature of Ethernet traffic"*, ACM SIGCOM, September 1993.

[17] T. Oetiker, D. Rand, *"Multi Router Traffic Grapher"*, http://people.eez.ethz.ch/~oetiker/webtools/mrtg/.

[18] P. Owezarski and N. Larrieu, ``*Coherent Charging of Differentiated Services in the Internet Depending on Congestion Control Aggressiveness*'', Computer Communications, special issue on ``Internet Pricing and Charging: Algorithms, Technology and Applications'', Vol. 26, issue 13, August 2003.

[19] K. Park, G. Kim and M. Crovella, ``*On the relationship between file sizes, transport protocols, and self-similar network traffic*'', IEEE ICNP, 1996.

[20] K. Park, G. Kim and M. Crovella, ``*On the Effect of Traffic Self-similarity on Network Performance*'', SPIE International Conference on Performance and Control of Network Systems, November, 1997.

[21] K. Park and W. Willinger, ``*Self-similar network traffic: an overview*'', In Self-similar network traffic and performance evaluation, J.Wiley & Sons, 2000.

[22] V. Paxson, G. Almes, J. Mahdavi, M. Mathis, *"Framework for IP Performance Metrics"*, RFC 2330, May 1998.

[23] V. Paxson, A. Adams, M. Mathis, "*Experiences with NIMI*", Passive and Active Measurements (PAM), 2000.

[24] V. Ribeiro, M. Coates, R. Riedi, S. Sarvotham, R. Baraniuk, *"Multifractal cross traffic estimation"*, ITC specialist seminar on IP traffic measurement, september 2000.

[25] V. Ribeiro, R. Riedi, R. Baraniuk, *"pathChirp: efficient available bandwidth estimation for network paths"*, Passive and Active Measurements (PAM'2003), 2003.

[26] J. Strauss, D. Katabi, F. Kaashoek, *"A measurement study of available bandwidth estimation"*, Internet Measurement Conference (IMC'2003), Miami, Florida, USA, October 27-29, 2003.

[27] D. Veitch, P. Abry, ``*A wavelet based joint estimator for the parameters of LRD, Special issue on Multiscale Statistical Signal Analysis and its Applications*'', IEEE Trans. Info. Th. April 1999, Volume 45, No.3, 1999.

[28] Y. Zhang, N. Duffield, V. Paxson, and S. Shenker, ``*On the Constancy of Internet Path Properties*'', Proc. ACM SIGCOMM Internet Measurement Workshop (IMW'2001), San Francisco, California, USA, November 2001.

16 AS: Autonomous System.