POLITECNICO DI TORINO
Repository ISTITUZIONALE

Multilevel bandwidth measurements and capacity exploitation in gigabit passive optical networks

Publisher copyright

(Article begins on next page)

04 August 2020

Multilevel Bandwidth Measurements and Capacity Exploitation in Gigabit Passive Optical Networks

Arianna Rufini, Marco Mellia, Edion Tego, Francesco Matera

A. Rufini, F. Matera, E. Tego, Fondazione Ugo Bordoni, via del Policlinico 147, 00161 Roma, mat@fub.it.
M. Mellia, Politecnico di Torino, Corso Duca degli Abruzzi 24, 10129, Torino (IT)

ABSTRACT

We report an experimental investigation on the measurement of the available bandwidth for the users in Gigabit Passive Optical Networks (GPON) and the limitations caused by the Internet protocols, and TCP in particular. We point out that the huge capacity offered by the GPON highlights the enormous differences that can be showed among the available and actually exploitable bandwidth. In fact, while the physical layer capacity can reach value of 100 Mb/s and more, the bandwidth at disposal of the user (i.e. either throughput at transport layer or goodput at application layer) can be much lower when applications and services based on TCP protocol are considered. In the context of Service Level Agreements (SLA) verification, we show how to simultaneously measure throughput and line capacity by offering a method to verify multilayer SLA. We also show how it is possible to better exploit the physical layer capacity by adopting multiple TCP connections avoiding the bottleneck of a single connection.

Index Terms—Fiber Networks; GPON; Quality of Service; TCP; Throughput, Goodput, Multisession

1. Introduction

The ever-increasing bandwidth-hungry applications bring Internet Service Providers (ISP) and industries to adopt new architectures based on Fiber To The Building (FTTB) and Fiber To The Home (FTTH) solutions and in particular by adopting Gigabit Passive Optical

Network (GPON) systems [1]. These new architectures provide ultra-broadband capacities but how and if such bandwidths will be really exploitable by end-user is still an open question, especially since the relationship among network quality (Quality of Service, QoS) and the user quality (Quality of Experience, QoE) can depend on different factors [2-3].

It is well known that the amount of useful bits for seconds (defined as either *goodput* if considered at application layer, or *throughput* if at transport layer) can differ very much with respect to the capacity offered at the physical layer, or line capacity [3-9]; for instance, in case of the Transfer Control Protocol (TCP) [10-12] the throughput can be much lower than the line capacity. This results particularly evident in case of network conditions with high values of bandwidth-delay product [5], which are especially common in high capacity accesses with line rate higher than 20 Mb/s [13]. This difference between line capacity and throughput is a cause of several problems related to Service Level Agreement (SLA) verification between customers and ISPs, since the ISP tends to consider the bandwidth in terms of line capacities while the customer tends to consider only the useful bandwidth (i.e. throughput). Therefore, from the SLA point of view, novel tools are needed to simultaneously evaluate throughput and line capacity, i.e., to verify both the effective bandwidth at disposal of the user and the one offered by the access technology used by the ISP [14]. On the other hand, from the bandwidth efficiency point of view, novel investigations are necessary both on new and on already well known methods to suitably exploit the enormous capacities offered by the optical fiber accesses, as for an instance the TCP multisession that has been tested in this paper [15].

The most reliable method to estimate the line capacity is to locate a specific measurement device at the gateway at the end user's home (modem, CpE,…). This method is clearly not scalable and it is very expensive, since it requires to install monitoring boxes very close to the customers' access links. Conversely, cheap methods to evaluate the user connection quality are based on "speed test" tools, which measure the time to download a specific data file from a dedicated server. Therefore such speed tests measure the application layer goodput (or the transport throughput) and they are not reliable to evaluate the line capacity. For instance, in [13] it was shown that the goodput is quite close to the line capacity in case of conventional ADSL2+ connections having a channel bit rate lower than 10 Mb/s. However, for line rate higher than that, the obtained throughput can be very different from the actual line capacity.

In this paper, we specifically consider the measurement of line rate and application goodput of PON access technologies. To the best of our knowledge, TCP performance over PON networks has been the subject of few papers [16-18], and these have all focused on IEEE EPON systems. For GPON systems, only [19] aims at assessing the impact that the GPON MAC layer may have on TCP. In [19] the authors report a preliminary study of TCP performance issues that may arise in GPON networks.

This paper has dual aim: first, to show methods to simultaneously measure throughput and physical line capacity in very wide broadband GPON accesses, and second, to illustrate how to overcome the throughput limitations due to TCP. To this extent, we present experimental tests that were carried out in a testbed consisting of a backbone core and an access (GPON) portions, representing a common wide regional network [20].

The remainder of the paper is structured in the following way: Sect. 2 reports a short overview on the bandwidth behavior in TCP and UDP measurement contexts, and on the QoS measurements. In Sect. 3 we describe the experimental setup and in Sect. IV we report the results for throughput versus line capacity for a GPON line at 100 Mb/s considering both TCP and UDP case. Sect. 4 is concluded by a proposal to implement speed tests to measure both the line capacity and the user throughput in such a way to certify the SLA between ISP and user, giving some tools to point out the user satisfaction from the application point of view. In Sect. 5 we experimentally test the multisession approach [6][15] that represents another method to measure the line capacity and overall to fully exploit the capacity in TCP environments. Section 6 reports some considerations on our methods when they operate in presence of impairments located in wide area networks segments and we report some tests carried out supposing a bandwidth bottleneck in the transmission between two routers. Finally, conclusions are drawn in Sect. 7.

2. OVERVIEW ON BANDWIDTH DEFINITION AND MEASUREMENT AND THE CONTEXT OF QUALITY OF SERVICE

Quality of Service (QoS) is the term that characterizes the performance from the network point of view and it is usually measured by means of some parameters as bandwidth, jitter, delay and packet loss.

QoS refers to a network's ability to achieve maximum bandwidth and deal with other network performance elements like latency, error rate and availability. Quality of service

also involves controlling and managing network resources by setting priorities for specific types of data (video, audio, etc.) on the network. More details on the QoS definition can be found for an instance in [2], that it has been the reference point for our investigations.

Currently, for QoS evaluation, one of the most critical parameter is the "bandwidth" that is also a subject of dispute between users and ISPs, and it is due to the fact that the bandwidth measured by the user depends on too many factors that are independently by the physical channel [3][13].In particular, for the bandwidth definition, we have to specify the OSI layer we refer to, distinguishing between line capacity and bandwidth at disposal of the user. Currently, the Internet services and applications, most notably HTTP, are based on IP protocol at the network layer, and most of them rely on TCP at the transport layer. This implies that download throughput is regulated by TCP. Theory and experimental tests [4-13] showed that TCP flow and congestion control mechanisms suddenly become the major bottleneck when exploiting high capacity paths with large Round Trip Time (RTT). In particular the download time of file depends on the Receiver Window (RWND) and on the Congestion Windows (CWND). The former depends on the Operating System (OS) installed by the user on its PC [4-5], while the latter depends on the congestion control implemented algorithms. We remember that the throughput, B_t, depends on these parameters, and in particular on the line capacity, B_c, according to the following relationship:

$$B_t \leq \frac{\min(CWND, RWND)}{RTT} \leq \min\left(B_c, \frac{RWND}{RTT}\right) \qquad (1)$$

In particular the sender is allowed to transmit no more than the minimum amount of data specified by CWND, RWND and RTT. As for any sliding window protocol, TCP throughput is proportional to the window size, and inversely proportional to the sender-receiver RTT. The mechanism of the TCP Congestion Control drives the CWND values to the actual path capacity that cannot be larger than the capacity of the bottleneck link. For example, in [13], it was shown that Windows XP (which is still used) suffered a strong reduction of the throughput with respect to the line capacity also in case of 20 Mb/s access as the one offered by ADSL2+ technology. Important improvements were showed up by OSes like Windows 7 and Linux, even though limitations were well observed in exploiting higher speed access capacities in presence of moderate RTT, cases that verify for accesses obtained by means of GPON.

Eq. (1) refers to throughput in stationary conditions; to know the detailed throughput time

evolution more complex equations are necessary and simulations can be useful. For instance in [22], Sect. 2.6, a simulation tool is reported to analyze the throughput behavior, and in particular to define the length of the packets to be used for throughput tests, especially for the case of long RTT.

The difference between goodput and line capacity could be deeply limited by avoiding flow control algorithms, such as by using UDP. Notably, UDP implements no retransmission for the case of packet loss, and does not perform congestion control. These simple considerations point out that the QoS should be defined for each Layer of OSI stack, and as a consequence we could define corresponding SLAs related to different OSI Layers, from physical to application Layer.

With the service evolution, the trend is to shift the SLA verification towards applications. This implies that metrics and measurements have to be applied at the application layer; therefore suitable SLA parameters must be defined for specific Web services such as YouTube, specified in terms of application-level measurable parameters (e.g. Mean Opinion Score [3]). These Application SLAs (ASLAs) will necessarily depend on lower-layer SLAs: for high-quality HD video delivery, it is necessary to have a very good large broadband physical connection that has to be guaranteed by a suitable SLA between ISP and user. This is a necessary but not sufficient condition for high QoS at application layer.

3. EXPERIMENTAL SETUP

The adopted testbed is shown in Fig.1 [3][20]. The core part is composed of four Juniper M10 routers interconnected using 1 Gb/s long haul optical links in the Rome area. Three Cisco 3845 edge routers are deployed at the access part of the network by means of Gigabit Ethernet optical links. Finally, the testbed is completed with GPON access networks composed of an OLT (Optical Line Termination) and up to eight Optical Network Units (ONUs), offering a shared bandwidth equal to 1.244 Gb/s. A desktop PC is connected to the user ONU. A high-end server is then connected using Gigabit Ethernet links. To emulate different RTT, a delay generator is added on the path between Server and Client to introduce an additional delay d.

To guarantee an End-to-End minimum bandwidth in the backbone path, we use the technique described in [20] that allows us to assign a guaranteed bandwidth between two end-points of the network by means of different tagging techniques, i.e., Virtual LAN and

Virtual Private LAN Service (VPLS). This technique enforces strict bandwidth requirements so that background traffic that may be present on the testbed does not interfere with our tests.

QoS analysis is carried out by exploiting the properties of IPERF tool [21].

We consider three different Operating Systems (OS) at the client, namely: Windows XP SP3, Windows 7, and Linux Ubuntu 9.10. The maximum CWND at the server was set to 512kB, which is larger than the actual maximum RWND imposed by default for all considered OSes.
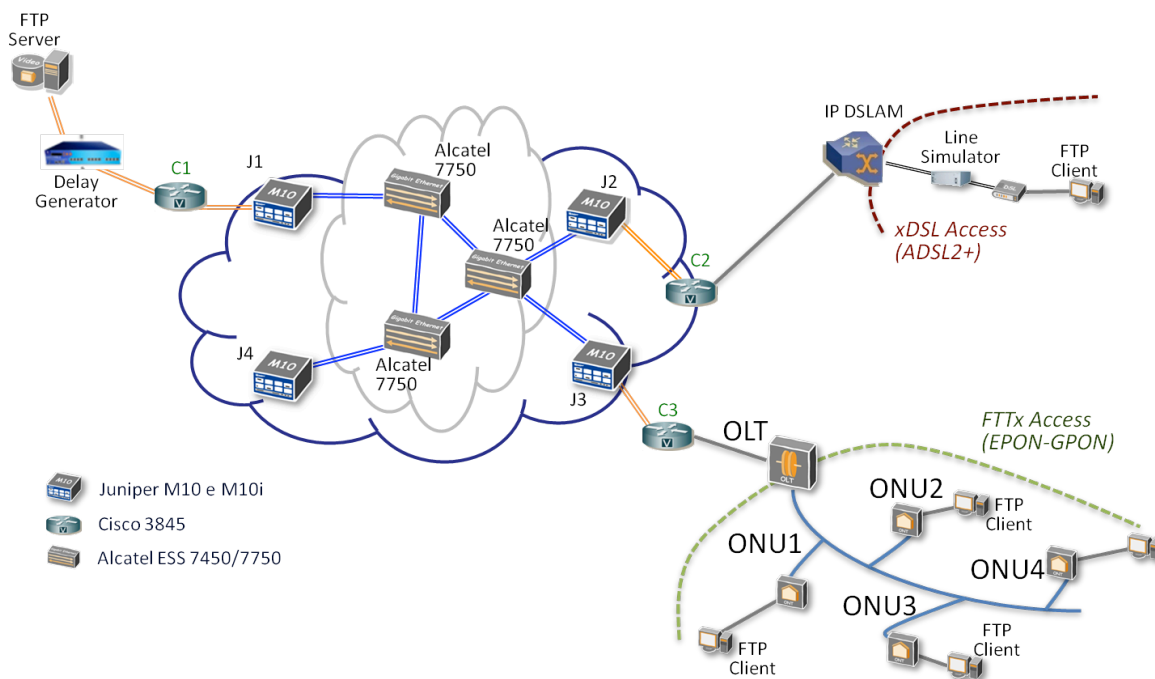

Fig. 1: Testbed design

4. EXPERIMENTAL RESULTS

In this section we report the results on the throughput behavior in GPON accesses distinguishing between TCP (a) and UDP (b).

a) TCP case

In fig.2 we report the throughput measured at time instant during a file transfer test.. Line capacity is set to 100 Mb/s. We report two curves corresponding to "No additional delay (d=0)" and introducing a delay equal to 100 ms in the network path between server and client, so RTT can be assumed equal to network delay, d. Linux is used as the operating system.
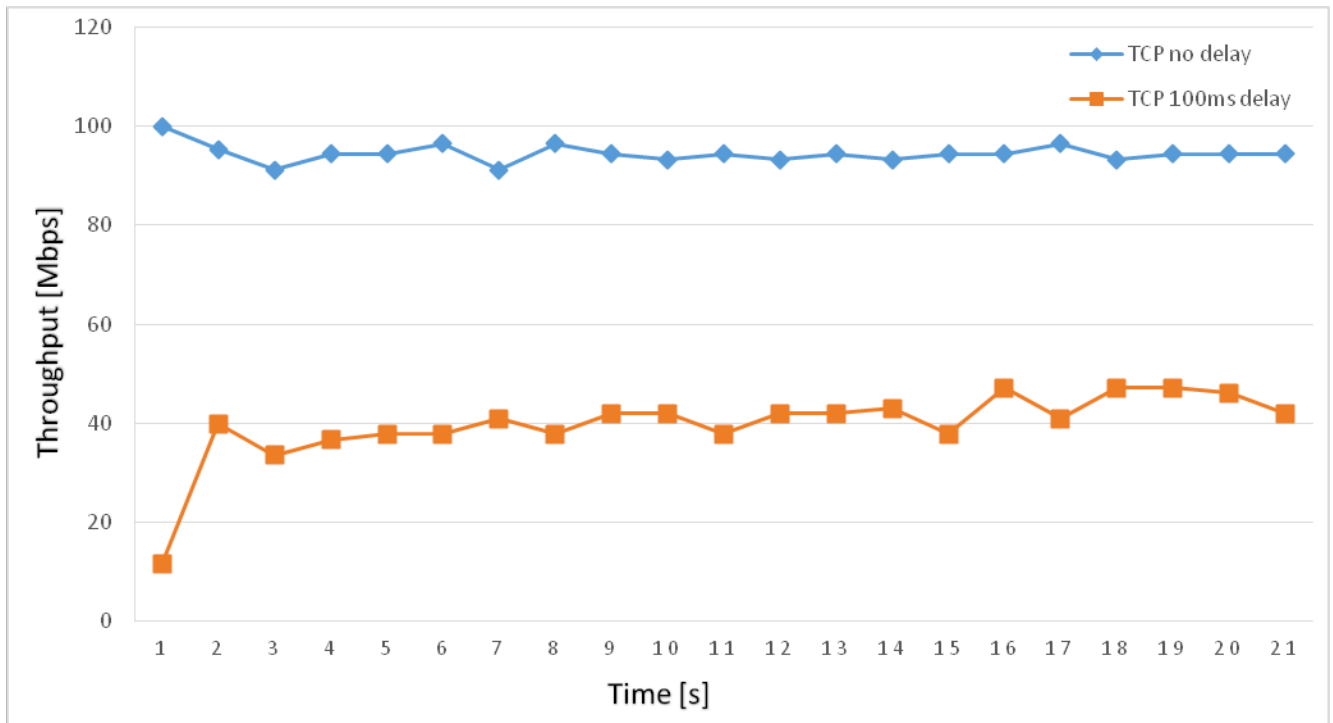
Fig. 2: Throughput measurements over time; line capacity of 100 Mb/s

For d=0 we see a modest fluctuating behavior due to the TCP congestion control algorithm: the TCP connection tries to exploit the maximum capacity of the line increasing the CWND; when the transmission bit rate overcomes the line capacity, packets may be lost, and TCP reacts by reducing CWND with a consequent rate decrease. This continuous increasing-decreasing behavior manifests with the small bandwidth fluctuations, with an average throughput equal to 94.5 Mb/s with a standard deviation of 2.3 Mb/s. We repeated this measurement 50 times obtaining for each realization an average throughput around 94.5 Mb/s with a variation less than 0.5%, confirming the same standard deviation.

With respect to flow with no additional delay, for RTT=100 ms the behavior is totally different and in particular we observe a typical transit behavior characterized by an initial fast growth of the throughput in time. This is due to the TCP slow start phase, when the congestion window grows exponentially, doubling every Round Trip Time until reaching the maximum value given by the RWND in our case. The obtained throughput is thus given by (1) as RWND/RTT. It has to be pointed out that the case of RTT=100 ms can be considered as a typical delay between server and client in a continental environment.

It has to be pointed out that RWND is an arbitrary value depending on the operating system implementation and its setting is critical because it influences the performance. In

fact if the threshold value is set too high relative to the network Bandwidth Delay Product (BDP), the exponential increase of congestion window generates many packets losses, with subsequent significant reduction of the connection throughput. On the other hand, if it is too small, TCP results in poor utilization especially when BDP is large.

In case of fig. 2, the O.S. sets RWND=512 Kbyte that corresponds to a throughput of about 95.5 Mb/s when no additional delay is added in the network path (blue line) and about 42 Mb/s for RTT=100 ms. The same value was obtained by repeating the measurement 50 times.

Fig. 2 clearly shows the limitations in terms of throughput due to the TCP behaviour and therefore the difficulties of TCP in exploiting huge line capacities as in case of fiber accesses under large RTT. This behavior is better summarized in fig. 3 which reports the throughput versus RTT for different OSes for a line capacity of 100 Mb/s. The experimental throughput is obtained as a time average along a period of 10 sec and each experimental point represents the average from 50 realization tests (for each experimental point the relative standard deviation was always lower than 1%). Evident substantial differences between Windows XP and more recent OSes (Windows 7 and Linux) are due to enhanced TCP algorithm implementation, related to adaptive parameters in the algorithm (i.e., Auto-Tuning of the RWND which can grow much larger than the 64kB offered by Windows XP).
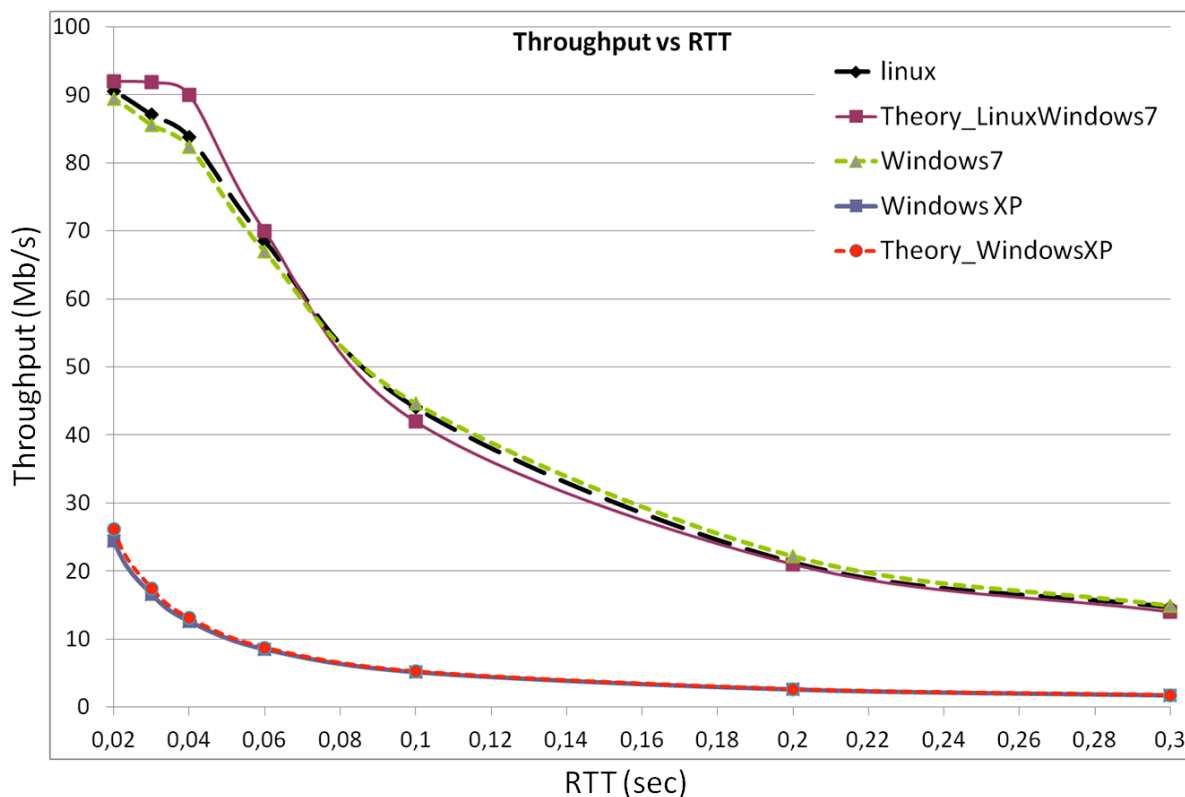

Fig. 3: Throughput vs RTT for different OSes

In the same figure we also report the analytical behavior from eq. (1) adopting Windows XP and Linux OSes, assuming as RWND 512 and 64 Kbyte for Linux and Window XP respectively. As illustrated in the next section, this strong bandwidth reduction could be avoided by adopting UDP protocol.

b) UDP case

UDP does not implement any sliding window algorithm that limit the throughput; therefore, if the server-client transmission is the same as the line capacity, the UDP throughput is maximum and equal to the line capacity (apart the overheads of the IP and lower layers encapsulation). This is shown in fig. 4 where we report the throughput versus time in the UDP case. We enforced the server to transmit data at 95.6 Mb/s. No loss was thus observed at the receiver. These tests were performed with an additional delay between server and client equal to 100 ms. In fig. 5 we detail the jitter behavior in the same test conditions of fig. 4; it has to be pointed out that such jitter values have no influence for current applications also based on HD and 3D video contents.
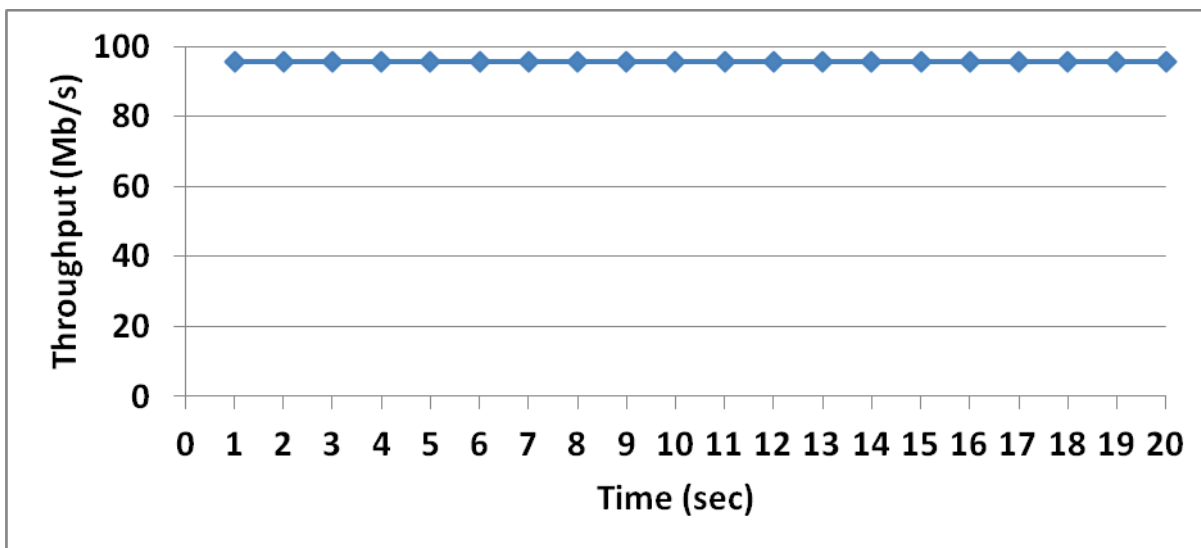

Fig. 4: Time evolution of the UDP Throughput for a line capacity of 100 Mb/s and RTT=100 ms
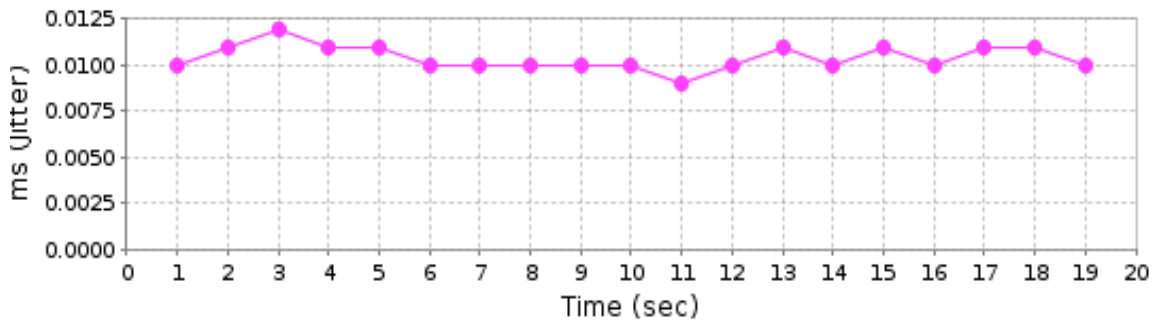
Fig. 5: UDP jitter for a line capacity of 100 Mb/s and RTT=100 ms

Since UDP does not implement congestion control mechanism, we expect to suffer packet losses when the server transmission rate exceeds the bottleneck capacity. This behavior is clearly shown in fig. 6 where we report the packet loss (in percentage with respect to the total packets) versus server data transmission (Tx Rate). As expected, loss occurs as soon as the data transmission rate from the server exceeds 95.5 Mb/s. Conversely when the bit rate is higher than the line capacity we observe packet losses corresponding to difference between channel bandwidth and server transmission rate. No difference was observed among Windows XP, W7 and Linux as far as the data loss behavior was concerned.
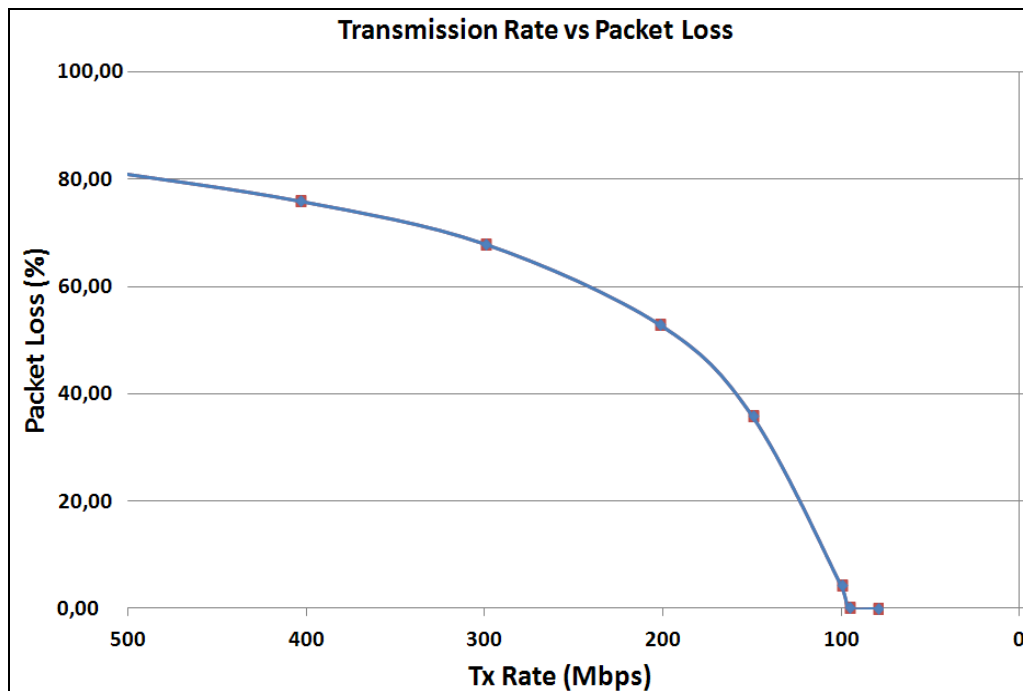

Fig. 6: Packet loss (%) versus server data transmission (Tx Rate) in case UDP with RTT=100 ms

c) Considerations and proposal for GPON bandwidth measurements

The results shown in this section suggest us a simple method to simultaneously measure the QoS according to different aspects related to the OSI layer and in particular to measure:

- the line capacity to verify SLA between the user and the ISP;
- the throughput (as available bandwidth for the user);
- the goodput (as available bandwidth for the user at application layer).

To perform this set of measurements, a software agent based on the following steps, has been developed. In particular the building blocks of the algorithm implemented by the agent are the following:

i) First, it performs throughput measurement by adopting a classic TCP file download method to evaluate B_t;

ii) It measures other QoS parameters such as delay t_d (RTT measured with PING), congestion window and threshold value;

iii) After the measurements of RTT and B_t, an estimation of the line capacity, B_c, can be obtained by means of eq. (1);

iv) Secondly, to measure the line capacity it performs a UDP test to obtaining other QoS parameters such as jitter and packet loss. UDP test is divided in two steps. First a UDP stream is sent from the server with a line capacity of B_c', measuring the lost datagrams with respect to the transmitted datagrams. In such a way we obtain the useful transmitted bits, BT_U, and from the total transmitted bits BT_T, we can achieve the estimated line capacity as $B_c = (BT_U/BT_T) * B_c'$. In the second step the effective line capacity is measured by means of a UDP test based on downloading of a file and forcing the transmission rate $B_c * = B_c - \varepsilon$ (i.e. $\varepsilon=0.001*B_c$), verifying that a packet loss is lower than 0.1%. The value of ε depends on the required reliability of the measurement and in particular on the SLA between ISP and user. If the required packet loss is verified $B_c *$ is the line capacity.

5. MULTISESSION TRANSMISSION

The previous section has shown that when the RTT is large, UDP could be more appealing than TCP, with the conditions to have a transmission bit rate lower than the line capacity. However, no congestion control would be available, thus making the usage of UDP

impractical, especially if we consider the case of a shared channel where packet losses can be caused by concurrent flows competing for the same capacity.

On the other hand, most services are based on TCP. Those would be strongly limited in case of large RTT, and some solutions have to be proposed to overcome the TCP limitations shown in the previous section to exploit all physical layer capacity.

Here we experimentally demonstrate the possibility of using a multisession transmission, where a file is subdivided into N files and each sub-file is transmitted at the same time with a bit rate $B_m = B_c/N$.
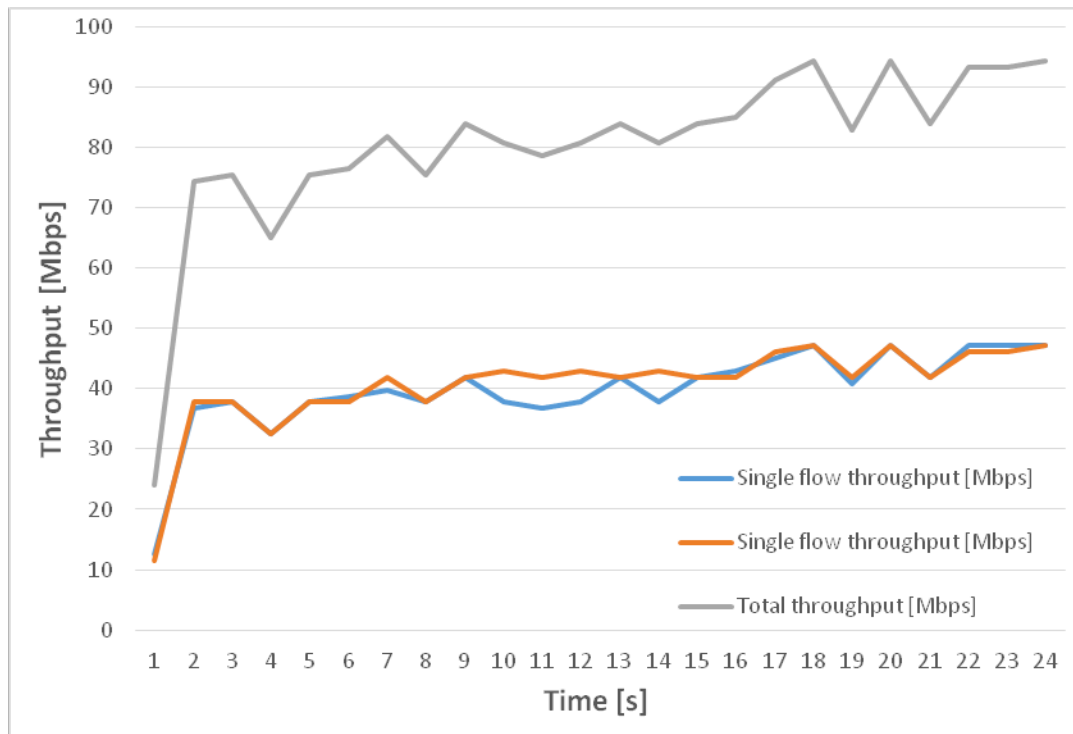

Fig. 7: Time evolution of the TCP Throughput for a line capacity equal to 100 Mb/s and RTT=100 ms in case of 2 flows

In fig. 7 we report the case of two flows. Comparing with fig. 2 we can see that the flows behave independently. After the initial transient, the throughput reaches a value around 42 Mb/s for each flow. In this case the multisession permits to much better exploit the line capacity doubling the throughput. However the sum of the flow rates is still lower than the line capacity.
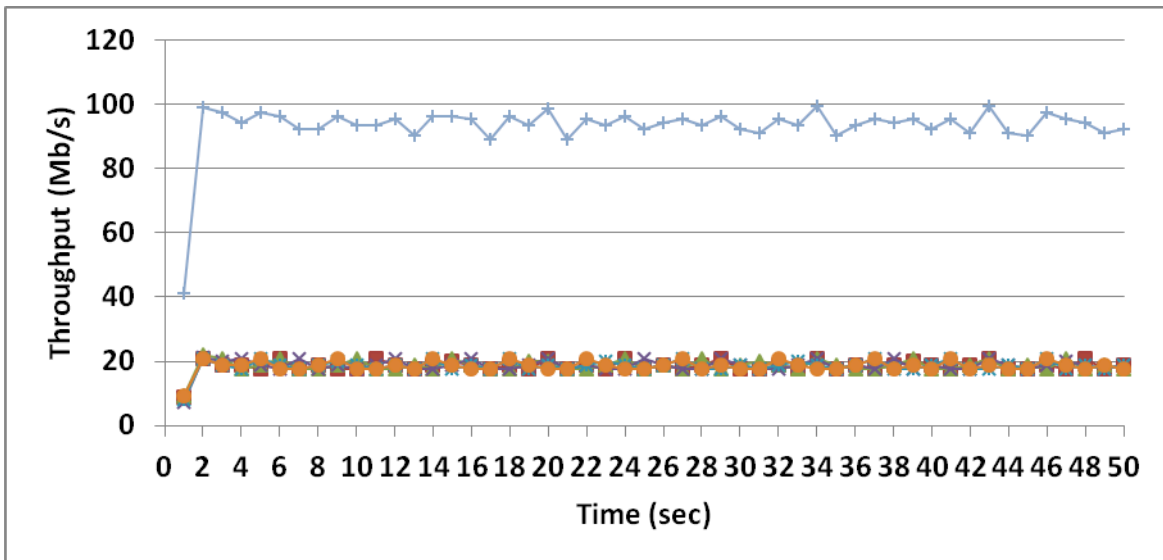
Fig. 8: Evolution over time of TCP Throughput for a Multisession with 5 flows. The highest curve is the sum of the five flows. Line capacity equal to 100 Mb/s and RTT=100 ms
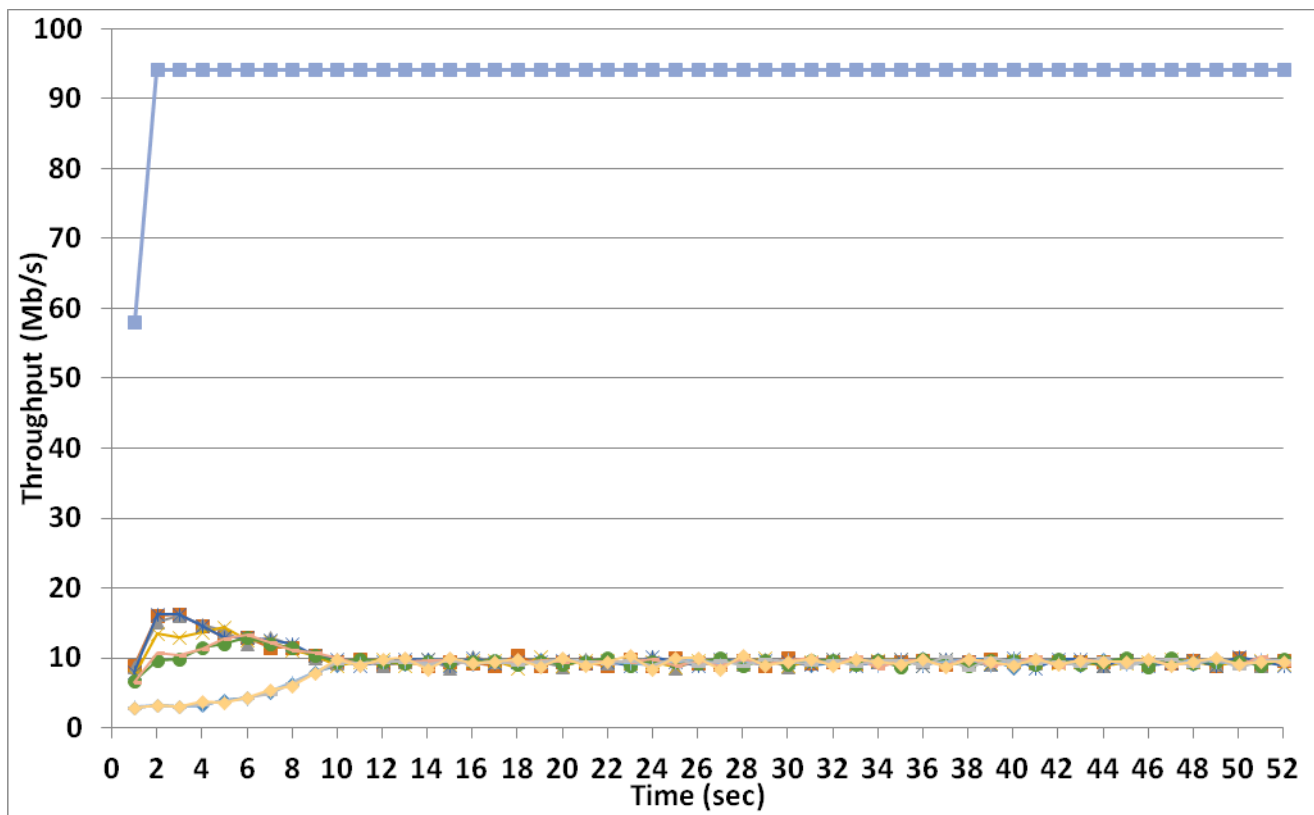

Fig. 9: As in fig. 8 but for a Multisession with 10 flows. The highest curve is the sum of the ten flows. Line capacity of 100 Mb/s and RTT=100 ms

In figures 8 and 9 we report the case of a multisession download using 5 TCP flows (each one obtaining approximatively a transmission rate of 20 Mb/s, up) and 10 TCP flows (each one about 10 Mb/s, bottom). As shown in these figures, the total throughput coincides with the maximum throughput (95.5 Mb/s) for a line capacity of 100 Mb/s. The measured throughput is lower than the line capacity because of the overhead introduced by layers of the network stack. As expected, the multisession download fully exploits the line capacity, and furthermore it can also be used to measure the line capacity by introducing a measurement based on N flows in TCP mode with bit rate B_c/N, where B_c is found in point iv in the method reported in Section IV.C. As shown by the comparison between fig. 8 and fig. 9 a reliable measurement of the line capacity can be obtained for values of N equal to 10 flows.

Clearly using multiple TCP flows all the line bandwidth tends to be exploited and this can have impact on delays, since the source would send more packets per unit of time, and create more congestion on the bottleneck buffer, increasing the end-to-end delay (and thus RTT). To see the impact of the multisession we run a RTT test, using Ping, while doing the bandwidth test. Results are reported in table 1, which, for each scenario, reports the minimum, average and maximum RTT. As expected, as soon as the sender is injecting traffic, the RTT grows (due to buffers filling up). We observe a small increase in the average RTT with the increase of TCP concurrent flows. While this can be a problem if widespread used among user to download content from the Internet, we believe it is acceptable in the context of bandwidth test which is seldom run and last for few seconds. Note that when the sender uses UDP (which does not implement congestion control), we see that the buffer is overloaded for most of the time (min RTT much higher). This brings to the problem of buffer sizing. Indeed, the buffer size (and the Active Queue Management eventually used) determines the packet loss probability, thus affecting TCP congestion control algorithms, thus possibly impairing the observed throughput.

Table 1: RTT values during TCP and UDP tests

	Ping + No flows	Ping + 1 TCP Flow	Ping + 2 TCP flow	Ping + 5TCP flows	Ping + 10 TCP flows	Ping + UDP flow
Min RTT	11.2ms	12.2ms	12.8ms	13.4ms	14.7ms	17.3ms
Average RTT	11.9ms	14.99ms	15ms	16.15ms	16.96ms	18.43ms
Max RTT	12.3ms	18.2ms	18.5ms	18.6ms	18.7ms	19.4ms

Buffer sizing is a controversial topic still in discussion among the research community, where either very small buffers [25] or very large buffers are suggested [26], with no clear winner. Assuming a droptail queue management, the golden rule to choose the buffer size is to set it according to the bandwidth delay product, i.e., $S \geq C*RTT$, where S is the buffer size, C is the end-to-end capacity [24]. In our case, we consider C=100Mb/s and RTT=100ms, which leads to $S \geq 1.25$Mbytes. This would guarantee that, in case one TCP connection is loading the link, eventual congestion would not cause any degradation of the throughput. Notice however that a proper setting of the buffer size requires knowing both C and RTT, which is clearly unfeasible in generic cases. Regarding to GPON ONU buffer we have set the maximum value around 9.9 Mb, that would give us a bandwidth after line coding more than 9.9 Gb/s [23]. However, given the target of our work (bandwidth estimation and verification in ISP networks), we would not have any control on the buffers in the real networks. We leave the study of buffer sizing and AQM policies impact for future work.

6. CONSIDERATION ON WIDE AREA NETWORKS

The results reported in the previous Sections refer to a lab environment, that mainly analyze the network access behavior. Further investigations should analyze some impairments that could be present inside the metro-core networks, including the traffic forwarding in routers and the interaction among different line loads. A deep study on this topic would require measurements on real wide area networks and this is not the subject of this paper. Simulations analysis could be useful and in particular the model reported in [22] could give important information. To complete our analysis we prefer to report an interesting experimental test that shows up important considerations on measured bandwidths reported in the previous Sections when different loads are present in the network, that can induce traffic congestions in some network segments. In particular, looking at the set-up of fig. 1, we simultaneously analyze the throughput behavior of ONU1 and ONU2 with two different traffic sent by the Server. In particular a 100 Mb/s UDP traffic was sent to ONU1, while 100 Mb/s TCP traffic, which was started before the UDP traffic, with multisession approach, was sent to ONU2. Furthermore in order to see the role of a core congestion we set a VPLS bandwidth path [20] of 157 Mb/s between routers J1 and J3. In such a way we could observe a sort of fighting between the two ONU traffics.

In such experimental condition in fig. 10 we report the throughput measured at ONU1 (up) and ONU2 (below).
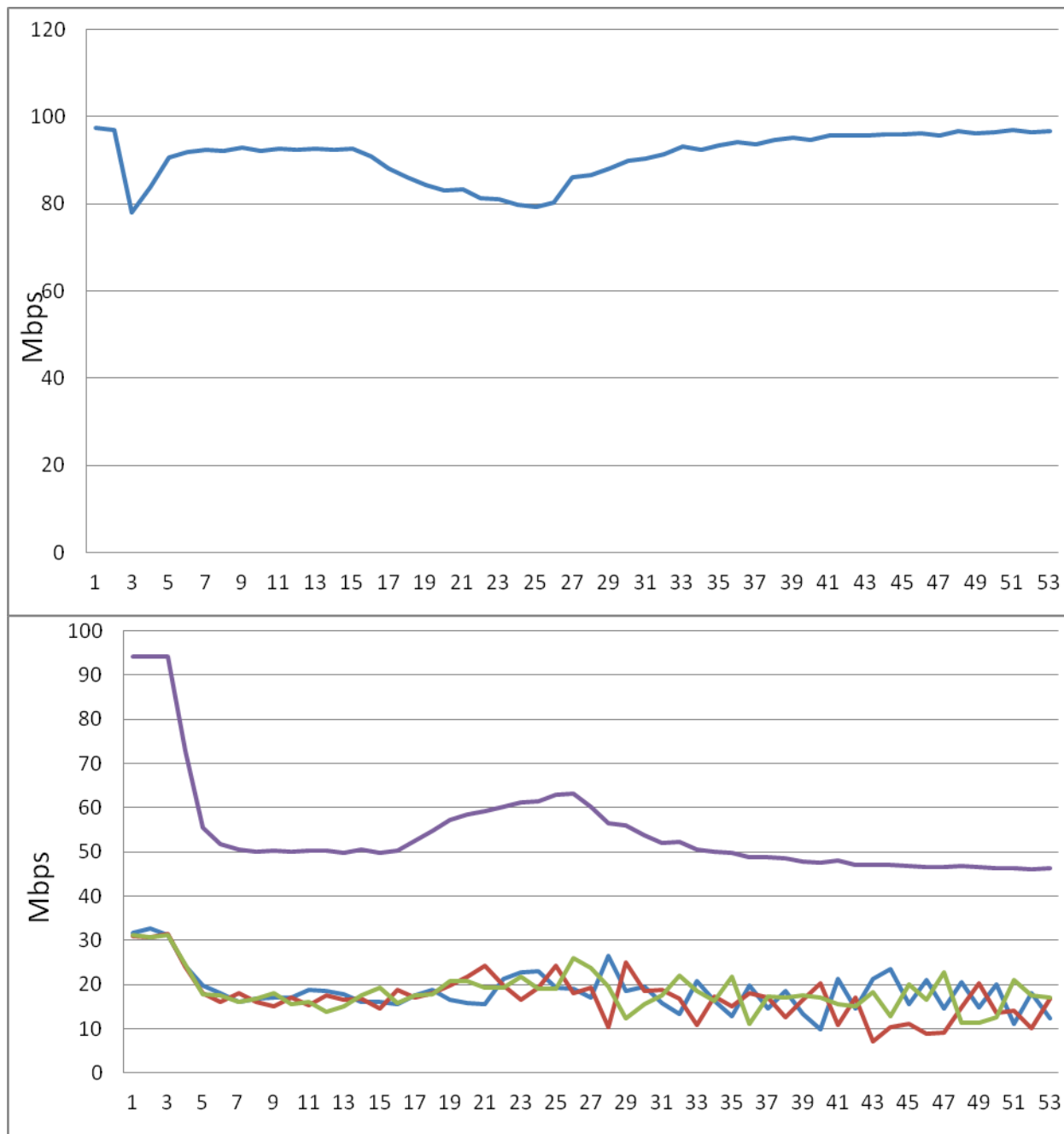

Fig. 10: UDP throughput at ONU1 (up) and TCP throughput (3 TCP parallel flows, in purple the sum of three flows) at ONU2 (below)

Figure 10 clearly shows that the throughput measurements strongly depend on all the network impairments, and therefore the interaction among UDP and TCP traffic can induce different bandwidth measurement, with consequence on the line capacity estimation. This

suggest us that correct monitoring of the access capacity would require also a passive monitoring of the traffic able to analyze some network environments as for example reported in [26].

7. CONCLUSION

The huge capacity offered by the optical fiber accesses introduce new challenges on the measurement and the exploitation of the bandwidth. In fact, due to the intrinsic behavior of the Internet protocols, the effective bandwidth at user disposal can be much lower than the line capacity. Furthermore the evaluation of the Quality of Service, and in particular the measurement of bandwidth requires further insights, since we have to distinguish to which OSI Layer we refer. The question is if we wish to measure either the line capacity or the available bandwidth at disposal of the user, since this has important consequences on the Service Level Agreement verification, especially between customers and their Internet Service Providers (ISPs).

In this work, we report an experimental investigation on the bandwidth measurement behavior in a GPON network, connected to a backbone area, analyzing the impact of the Internet protocols. In particular our tests showed that the available bandwidth can be much smaller than the line capacity when TCP is adopted. We show how UDP can allow to fully exploit the GPON capacity and we propose a method that allows to measure with few steps both the throughput and the line capacity permitting to have a tool to verify SLA between ISP and customer. We also show how to fully exploit the line capacity in a TCP environment by means of the multisession approach, that can be also adopted to evaluate the line capacity. Furthermore we also analyze the throughput behavior when bandwidth limitations are present in some core network segments. The obtained results clearly show that the throughput measurements strongly depend on all the network impairments. This suggest that correct monitoring of the access capacity would require also a passive monitoring of the traffic able to analyze some network environments. Future works will be based on monitoring of the traffic performed by passive measurements.

REFERENCES

[1] ITU-T Recommendation G. 984.1, "Gigabit-capable Passive Optical Network (GPON): general characteristics", March 2003.

[2] ETSI EG 202 057-4, "User related QoS parameter definitions and measurement", October 2005

[3] Valenti, A., Rufini, A., Pompei, S., Matera, F., Di Bartolo, S., Da Ponte, C., Del Buono, D., Tosi Beleffi, G.M. "QoE and QoS Comparison in an Anycast Digital Television Platform Operating on Passive Optical Network", Networks 2012 - Roma – October 15-18, 2012.

[4] Lakshman, T. V., Madhow, U., and Suter, B., "TCP/IP performance with random loss and bidirectional congestion", IEEE/ACM TRANSACTIONS ON NETWORKING, 2000, 8, (5), pp. 541-555.

[5] T. V. Lakshman, Member, "The Performance of TCP/IP for Networks with High Bandwidth-Delay Products and Random Loss", IEEE, and Upamanyu Madhow IEEE/ACM TRANSACTIONS ON NETWORKING, 1997. 5, (3), pp. 336-350.

[6] Mellia, M., Lo Cigno R., Neri, F., "Measuring IP and TCP behavior on edge nodes with Tstat", Computer Networks, 2005, 47, (1), pp. 1-21.

[7] Bolletta, P., Rea, L., "Monitoring of the User Quality of Service: Network Architecture for Measurements and role of Operating System with consequences for optical accesses", ONDM 2011 – Bologna, Italy February 2011.

[8] Del Grosso, A., Rea, L. Bolletta, P., Valenti, A., Luisi, A. M., "On the Impact of Operative Systems Choice in End-user Bandwidth Evaluation: Testing and Analysis in a Metro-access Network", IARIA ACCESS 2010 Sept. 2010 Valencia, Spain.

[9] Jansen S., and McGregory, A., "Measured Comparative Performance of TCP Stacks", PAM Conference, Boston (USA) March 31-April 1, 2005.

[10] Postel, J., Reynolds, J., "File Transfer Protocol", STD 9, RFC 959, October 1985.

[11] Postel, J., "Transmission Control Protocol", RFC 793, September 1981.

[12] King, J., "Parallel FTP Performance in a High-Bandwidth High-Latency WAN", SC2000, November 2000.

[13] Rea, L., Mammi, E., "Italian QoS Monitoring network: impact on SLA control", Networks 2012, Rome October 15-18, 2012.

[14]   MPLANE deliverable D1.1., http://www.ict-mplane.eu/

[15]   Tullimas, S., Nguyen, T., Edgecombe, R., "Multimedia Streaming Using Multiple TCP Connections", Journal ACM Transactions on Multimedia Computing, Communications, and Applications (TOMCCAP), vol. 4 issue 2, 2008, art. N. 12.

[16]   Chang, K.C., Liao, W., "On the throughput and fairness performance of TCP over Ethernet passive optical networks", IEEE Journal on Selected Areas in Communications 24, 3–12 (2006).

[17]   Chang, K.C., Liao, W., "TCP fairness in Ethernet over passive optical networks (EPON)", In: Proceedings of IEEE CCNC, Las Vegas, pp. 740–744 (2006).

[18]   Ohara, K., Miyazaki, N., Tanaka, K., Edagawa, N., "Fairness of downstream TCP throughput among diversely located ONUs in a GE-PON system", In: Proceedings of the Optical Fiber Communication Conference (OFC) (2006).

[19]   Orozco, J., Ros, D., "TCP Performance over Gigabit-Capable Passive Optical Networks", ICST Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering 2009.

[20]   Valenti, A., Pompei, S., Rea, L., Matera, F., Tosi Beleffi, G., Curti, F., Di Bartolo, S., Incerti, G., Forin, D., "Experimental Investigation of Quality of Service in an IP All-Optical Network Adopting Wavelength Conversion", IEEE/OSA J. of Optical Communications and Networking, 2009, 1, 2, pp. 170-179.

[21]   http://code.google.com/p/iperf/

[22]   Rufini, A. Multi - Service Level Agreement in Wired and Wireless Access Networks Dottorato di Ricerca in Ingegneria dell'Informazione e della Comunicazione - Sapienza Università di Roma - XXV Ciclo; http://www.fub.it/content/libri

[23]   Skubic, B., Chen, J., Ahmed, J., Wosinska, L., Mukherjee, B. "A Comparison of Dynamic Bandwidth Allocation for EPON, GPON, and Next-Generation TDM PON", IEEE Communication Magazine, March 2009, pp. S40-S48.

[24]   Wischik D., McKeown N. "Part I: buffer sizes for core routers", SIGCOMM Comput. Commun. Rev. 35, 3 (July 2005), pp. 75-78.

[25]   Guido Appenzeller, Isaac Keslassy, and Nick McKeown, "Sizing router buffers", SIGCOMM Comput. Commun. Rev. 34, 4 (August 2004), 281-292.

[26] Amogh Dhamdhere and Constantine Dovrolis, "Open issues in router buffer sizing", SIGCOMM Comput. Commun. Rev. 36, 1 (January 2006), 87-92.

[27] Traverso, S., Tego, E., Kowallik, E., Mellia, M., Matera, F., "Exploiting Hybrid Measurements for Performance Troubleshooting", submitted to Networks 2014, Madeira (Pt) September 2014.